# From continuous to discrete

## Global grids, auxiliary spheres and the cost of distortion

*The fifth monograph of the cartographic distortion series, following* No Map Is Best: A Synthesis of the Cartographic Distortion Programme

Silvano Salvador

# Contents

**Preface and Status of This Volume** **xxi**

**I The Global Discrete Frame** **1**

**1 Why Discrete: From the Limits of Continuous Projections to DGGS** **2**

1.1 The problem inherited from Monographs 1–4 . . . . . 2

1.2 The reference case: Mercator and rectangular cells . . . . . 2

1.3 Five open-source implementations compared . . . . . 3

1.4 What Chapters 2–9 do with this frame . . . . . 4

**2 The Graticule as the First DGGS** **5**

2.1 Scope: a discrete scheme with no hexagons, triangles, or code . . . . . 5

2.2 Marinos, Ptolemy, and the first documented coordinate grid . . . . . 5

2.3 The grid as Ptolemy specifies it . . . . . 6

2.4 What the graticule gets right, and what every later DGGS chapter is about . . . 7

2.5 Closing: the graticule is not a straw man . . . . . 7

**3 Transverse Mercator Zones as a Historical Precedent for Regional Distortion Optimization** **8**

3.1 Scope: zoning, not tessellation . . . . . 8

3.2 From Lambert and Gauss to Krüger: the ellipsoidal transverse Mercator . . . . 9

3.3 The zone as a distortion-control device . . . . . 9

3.4 An American instance: the State Plane Coordinate System . . . . . 10

3.5 How far a zone travels: a partial map of adoption . . . . . 10

3.6 Closing: the same trade, a different tessellation . . . . . 12

**4 The Military Grid Reference System: Alphanumeric Indexing on Top of a Continuous Projection** **13**

4.1 Scope: from zoning to addressing . . . . . 13

4.2 The problem MGRS was built to solve: too many incompatible grids . . . . . 14

4.3 From numeric coordinates to alphanumeric addresses . . . . . 14

4.4 What MGRS does not solve: junctions and the limits of zoning . . . . . 15

**5 Discrete Addressing Without a Polyhedron: Geohash and Open Location Code 17**

5.1 Scope: the seventh scheme, and the one that skips a step . . . 17

5.2 Geohash (2008): Morton order, reinvented and misattributed to no one . . . 17

5.3 Open Location Code (2014): the same graticule bisection, renamed . . . 18

5.4 The shared property: latitude/longitude bisection, not sphere subdivision . . . 19

5.5 Why both systems accept this, and what it means for this volume's frame . . . 19

**6 Web Mercator: The Efficiency/Accuracy Trade-off in a Projection Everyone Uses 21**

6.1 Scope: the trade-off this volume usually poses for DGGS, posed instead for a continuous projection . . . 21

6.2 What Web Mercator is: a conformal sphere trick wearing an ellipsoidal datum . 22

6.2.1 The EPSG registry's own initial refusal . . . 22

6.3 Why it was adopted anyway: square tiles, not scientific accuracy . . . 22

6.4 The area-inflation cost, quantified qualitatively . . . 23

**7 Compromise Projections Before Peters 25**

7.1 Scope: a cartographic tradition, not a single controversy . . . 25

7.2 Van der Grinten (1898/1904): the first widely adopted compromise . . . 25

7.3 Robinson (1963): compromise by eye rather than by equation . . . 26

7.4 Winkel Tripel (1921/1998): compromise by explicit averaging . . . 27

7.5 Closing: compromise as a third strategy, not a fallback . . . 27

**8 Interrupted and Composite Projections: Discontinuity as a Fifth Strategy 28**

8.1 Scope: splitting the globe on purpose . . . 28

8.2 From Mollweide interrupted to homolosine: Goode's 1916–1923 sequence . . . 28

8.3 Interruption as a general strategy: from globe gores to myriahedral maps . . . 29

8.4 Closing: discontinuity as this volume's fifth historical strategy . . . 30

**9 A Conformal Square, a Century Early: Peirce and Guyou 31**

9.1 Scope: seamless tiling as a nineteenth-century conformal result . . . 31

9.2 Peirce (1879): a square built by Schwarz–Christoffel mapping . . . 31

9.3 Guyou (1887) and Adams (1925): the same formula, reoriented . . . 32

9.4 A quantified distortion budget, a century before this volume's own . . . 32

9.5 Closing: a precedent this volume's polyhedral chapters do not supersede . . . 33

**10 Straight Lines That Are Actually Shortest: Azimuthal Projections for Navigation and Aviation 34**

10.1 Scope: a functional, not a mathematical, history . . . 34

10.2 The gnomonic projection: straight lines as shortest paths, from Renaissance globes to Lindbergh's chart . . . 35
10.3 Azimuthal equidistant: distance-true from one point, and the United Nations flag 36
10.4 Closing: a fourth strategy, orthogonal to the volume's usual axis . . . 37

**11 Reproducing the Five-DGGS Comparison: Normalized Area and Compactness 38**
11.1 Closing the todobox of Chapter 1 . . . 38
11.2 Normalized area: how uniform are the cells? . . . 38
11.3 Compactness: how close to regular is the average cell? . . . 39
11.4 Cross-check: does the reproduced data pass an independent sanity test? . . . 40
11.5 What remains open . . . 40

**12 An Independent Reproduction: H3 Geometry from Scratch 42**
12.1 Scope: item (i) of the previous chapter's todobox . . . 42
12.2 Results and comparison to the transcribed table . . . 43
12.3 What remains open . . . 44

**13 Does the H3/S2 Divergence Persist? The Full Resolution Range 45**
13.1 Scope: item (ii) of the reproduction chapter's todobox . . . 45
13.2 The extended table . . . 45
13.3 What remains open . . . 46

**14 The S2 Resolution-to-Level Ambiguity, Systematically 47**
14.1 Scope: returning to a question left open . . . 47
14.2 Mechanism one: date-line-crossing removal . . . 47
14.3 Mechanism two: a reverse-fit polar-cap threshold . . . 48
14.4 Follow-up: face-relative coordinates instead of latitude . . . 49
14.5 What this narrows, and what it still leaves open . . . 50
14.6 A third candidate: Kmoch et al.'s own processing code, located and tested . . . 50

**15 A Second Independent Reproduction: DGGRID Geometry from Scratch 52**
15.1 Scope: the DGGRID half of the previous chapter's open item . . . 52
15.2 Results and comparison to the transcribed table . . . 53
15.3 What remains open . . . 54

**16 A Third Independent Reproduction: OpenEAGGR Geometry from Scratch 55**
16.1 Scope: the last of the five schemes . . . 55
16.2 The growth factor: from aperture 7 at resolution 1 to aperture 3 in the limit . . . 56

16.3 Area and compactness: results and comparison to the transcribed table . . . . . 57
16.4 What remains open . . . . . 58

**17 Testing the Resolution-Index Reading: Does an Integer Refinement Shift Explain the Gap? 60**
17.1 Scope: turning a plausible reading into a checked claim . . . . . 60
17.2 The test . . . . . 60
17.3 What this does and does not settle . . . . . 61

**18 Testing the Catastrophic-Cancellation Guess with Synthetic Hexagons 62**
18.1 Scope: isolating one candidate mechanism . . . . . 62
18.2 Results: the gap does not grow as cells shrink . . . . . 63
18.3 What this does and does not settle . . . . . 64

**19 The Projections Behind the Grids: Snyder, HEALPix, and the Quadratic Cube 65**
19.1 Scope: opening the black box . . . . . 65
19.2 The Snyder equal-area icosahedral projection (ISEA) . . . . . 65
19.3 The rHEALPix projection: cylinder, cap, and a rearranged cube . . . . . 66
19.4 The S2 quadratic transform: uniformity bought, not guaranteed . . . . . 67
19.5 Synthesis: three answers to one cartographic question . . . . . 68

**20 Why a Cube: S2 Against a Longer Cube-Sphere Tradition 70**
20.1 Scope: a question the synthesis table left open . . . . . 70
20.2 A cube-sphere lineage that predates S2 by three decades . . . . . 70
20.3 What the earlier lineage solved that S2 does not . . . . . 71
20.4 Why a cube at all: the quadtree constraint . . . . . 71
20.5 Closing: two lineages, two design priorities . . . . . 72

**21 Orienting the Polyhedron: A Second, Independent Cartographic Choice 73**
21.1 Scope: a design axis this volume has not yet touched . . . . . 73
21.2 Before optimization: Fuller's eye and an inconsistent legacy . . . . . 73
21.3 Orientation as a three-parameter optimization problem . . . . . 74
21.4 Why this matters for a tessellation, not just a flat map . . . . . 75
21.5 Synthesis: two independent axes, not one . . . . . 75

**22 The Polyhedral Tradition Before and After Fuller 77**
22.1 Scope: Fuller's map was not the first, and the icosahedron was not the only choice 77
22.2 Cahill's octahedron, 1909–1913 . . . . . 77

22.3 A 1932 exchange, and the octahedron's persistence . . . 78
22.4 After Cahill: refining the same octahedron, twice . . . 78
22.5 Why the icosahedron eventually displaced the octahedron for this volume's purposes . . . 79
22.6 Synthesis: a two-hundred-year design space, of which this volume documents one corner . . . 80

**23 Confirming H3's Orientation: The Fuller Correspondence, Verified 81**
23.1 Scope: closing one branch of Chapter 21's open question . . . 81
23.2 H3's twelve pentagons: zero vertices on land . . . 81
23.3 Not any of Barnes's four optimized rows . . . 82
23.4 A match to Fuller, to within metres . . . 82
23.5 What remains open . . . 83

**24 The Modifiable Areal Unit Problem, Rediscovered on a Grid 84**
24.1 Scope: a problem imported from human geography, not derived here . . . 84
24.2 The classical formulation: two problems, one cause . . . 84
24.3 Does a DGGS abolish the problem, or just relocate it? . . . 85
24.4 Reframing this volume's own results as instances of the two problems . . . 86
24.5 A cartographic, not only statistical, restatement of MAUP . . . 87
24.6 Synthesis: relocated, not resolved . . . 87

**25 Aperture and Hierarchy: A Third Design Axis 89**
25.1 Scope: two axes fixed, a third left untouched . . . 89
25.2 What an aperture is, and where its name comes from . . . 89
25.3 Aperture 7: symmetric, but rotated . . . 90
25.4 Congruent and non-congruent hierarchies . . . 90
25.5 Pure and mixed aperture sequences . . . 91
25.6 Synthesis: a third axis, and where it leaves this volume's map . . . 91

**26 A Standard Written After the Fact: The OGC DGGS Abstract Specification 92**
26.1 Scope: naming a category this volume has assumed all along . . . 92
26.2 What the standard actually requires . . . 92
26.3 The four-implementation audit: mostly non-compliant, by the standard's own criteria . . . 93
26.4 A standard written after its own subject matter existed . . . 94

**27 The Auxiliary Sphere Behind Every Global DGGS 95**
27.1 Scope: a step taken before the projection is ever applied . . . 95

27.2 Why a sphere at all: the auxiliary-latitude idea . . . 95
27.3 The authalic sphere, concretely . . . 96
27.4 Documented practice: DGGRID's own stated construction . . . 96
27.5 A cautionary counter-example: when the auxiliary sphere is not authalic . . . 97
27.6 Synthesis: the sphere before the sphere . . . 97

**28 The Auxiliary Latitudes, in Formula 98**
28.1 Scope: from prose to equation . . . 98
28.2 Six latitudes, one ellipsoid . . . 98
28.3 Which of the six close, and which do not . . . 99
28.4 Why this matters for a volume about discrete grids . . . 100

**29 The Auxiliary Latitudes, Evaluated: How Much the Substitution Costs 101**
29.1 Scope: from equation to number, and to a Tissot of the step itself . . . 101
29.2 The six latitudes, evaluated: how far the substitution moves a point . . . 102
29.3 The step itself, in local distortion: authalic, rectifying, conformal . . . 102

**30 Tissot's Indicatrix Revisited: From Cartographic Ellipse to Bi-Lipschitz Constant 105**
30.1 Scope: an object cited, never examined . . . 105
30.2 Tissot's own construction, 1859–1881 . . . 105
30.3 The quasiconformal afterlife . . . 106
30.4 From ellipse to bi-Lipschitz constant . . . 106
30.5 What this reformulation is not . . . 107
30.6 Closing the loop back to Chapter 1 . . . 108

**31 The Indicatrix, Computed: Local Distortion in S2 and rHEALPix 109**
31.1 Scope: from the tool revisited to the tool applied . . . 109
31.2 Method . . . 109
31.3 rHEALPix: exactly equal-area everywhere, unevenly shaped . . . 110
31.4 S2: distortion of the actual cell grid, not the raw cube face . . . 111
31.5 Synthesis: two different shapes of the same problem . . . 111

**32 The Corner Reversal, Derived 113**
32.1 Scope: closing the open todobox item . . . 113
32.2 Method: the composite map in closed form . . . 113
32.3 The face-crossing angle, in closed form . . . 114
32.4 A genuine feature, for contrast: the interior area-scale minimum . . . 115
32.5 What this changes in Chapter 31 . . . 115

**33 The Ellipsoid, Not Assumed Away: rHEALPix's Indicatrix on WGS84** **117**
33.1 Scope: closing a gap left open twice . . . 117
33.2 The equatorial belt: distortion grows, in both directions, away from the equator 118

**34 Before Milnor: Euler, Gauss, and Chebyshev on Distortion** **120**
34.1 Scope: inevitability is not optimality . . . 120
34.2 Euler, 1775: no exact-scale map of any part of a sphere . . . 120
34.3 Gauss, 1827: the sphere is the rule, not the exception . . . 121
34.4 Chebyshev, 1856: optimal, not merely inevitable, for conformal maps . . . 121
34.5 Two centuries, two kinds of theorem . . . 122

**35 From Indicatrix to Score: A Genealogy of the Cost of Distortion** **123**
35.1 Scope: what the earlier chapters left as a pointwise measure . . . 123
35.2 Tissot's indicatrix as a starting point, not an endpoint . . . 123
35.3 Peters, 1983: distortion reframed as a normative cost . . . 124
35.4 Toward a combinable measure: Laskowski's distortion spectrum . . . 124
35.5 Goldberg and Gott, 2007/2008: six errors, one score . . . 125
35.6 2021: driving the score down, and what stayed unchanged . . . 125
35.7 The same move made for discrete grids . . . 126
35.8 Closing: an unfinished genealogy, not a solved problem . . . 127

**36 Closing the Gap: The Indicatrix, Derived, for H3** **128**
36.1 Scope: the gap left open by Chapter 31 . . . 128
36.2 The closed-form solution . . . 128
36.3 Verification: closed form against finite differences, face against face . . . 129
36.4 An empirical cross-check against real H3 cells . . . 130

**37 Dutton's Quaternary Triangular Mesh: A DGGS Before the Acronym** **131**
37.1 Scope: an instrument used, not examined . . . 131
37.2 Origins: a data-quality problem, not a tessellation problem . . . 131
37.3 The geometry: octahedron, quaternary subdivision, ZOT . . . 132
37.4 Facet numbering, geocodes, and the attractor idea . . . 132
37.5 QTM in the later DGGS landscape . . . 133
37.6 An independent geometric reproduction, and a numeric check of the attractor claim . . . 133
37.7 What Dutton's own framing adds to this volume's argument . . . 135

**38 The Hierarchical Triangular Mesh: The Same Octahedron, Independently, from the**

**Sky** **137**

38.1 Scope: a second octahedron, from an unconnected literature . . . 137

38.2 Origins: range queries on the sky, not precision on the ground . . . 137

38.3 The geometry: octahedron, aperture-4, trixels . . . 138

38.4 Shape statistics, as HTM's own literature reports them . . . 138

38.5 What the convergence, and not just the resemblance, suggests . . . 139

**39 An Independent Reproduction: HTM Geometry from Scratch** **140**

39.1 Scope: closing the item Chapter 38 left open . . . 140

39.2 A caught error, stated rather than hidden . . . 140

39.3 Results: three published figures, checked independently . . . 141

39.4 Where this sits in the volume's wider comparison . . . 141

**40 Closing the Gap, Differently: The Indicatrix Attempted, and Found Wanting, for HTM** **143**

40.1 Scope: why HTM cannot simply reuse Chapter 36's method . . . 143

40.2 Two bugs caught in building the comparison . . . 144

40.3 Near the centre: rough agreement . . . 144

40.4 Farther out: area shrinks, not grows . . . 145

**41 The Indicatrix, Side by Side: A Cross-Grid Synthesis** **147**

41.1 Scope: six chapters, one comparison never assembled . . . 147

41.2 The six figures, collected . . . 147

41.3 What the six have in common, and where they genuinely differ . . . 147

**42 Not an Artefact of One Octant: The HTM Transect, Extended** **151**

42.1 Scope: the second item of Chapter 40's todobox . . . 151

42.2 No ties, and an unplanned symmetry finding . . . 151

42.3 Octant invariance . . . 152

42.4 Near the centre: the non-convergence, at two independent azimuths . . . 152

42.5 Farther out: area shrinkage, confirmed at both azimuths and every octant . . . 153

**43 Between the Two Points: Sweeping the Fundamental Domain** **154**

43.1 Scope: the first item of Chapter 42's todobox . . . 154

43.2 Results . . . 154

43.3 What this rules in, and what remains open . . . 155

**44 One Hundredth of a Degree: Locating the Crossing** **156**

44.1 Scope: the first item of Chapter 43's todobox . . . 156
44.2 Results . . . 156
44.3 What this closes . . . 157

**45 Generalized Balanced Ternary: An Addressing Algebra Seven Years Before QTM 158**
45.1 Scope: a digit assignment cited, never examined . . . 158
45.2 Origins: pattern recognition and spatial data processing, not cartography . . . 159
45.3 The geometry: seven digits, one addition table, multiple carries . . . 159
45.3.1 Independent check of two of van Roessel's numeric claims . . . 160
45.4 From planar addressing to icosahedral DGGS: the twenty-five-year gap . . . 161
45.5 The missing direction: from Cartesian coordinates back to a GBT address . . . 162
45.6 What GBT's own framing adds to this volume's argument . . . 164

**46 An Independent Reproduction: GBT's Addition Algebra from Geometry Alone 165**
46.1 Scope: the open item left by the previous chapter . . . 165
46.2 Deriving the addition table from geometry, and checking it . . . 165
46.3 An exact planar baseline for this volume's IPQ comparisons . . . 166
46.4 Closing item (i): an independent reverse map, by a different route . . . 167

**47 Beyond Aperture Seven: An Independent Check of Sahr's Modified GBT 170**
47.1 Scope: item (ii) of the previous chapter's todobox . . . 170
47.2 MGBT's two alternating digit tables . . . 170
47.3 A second, exact relationship the patent does not itself state . . . 171
47.4 Consequence for this volume's isoperimetric-quotient table . . . 172
47.5 The icosahedral base-tile tables: transcription and consistency checks . . . 172
47.6 Revisiting item (ii): the corrupted exponent, resolved, and its consequence . . . 173

**48 A Third Pre-2000 Family: White's Diamond Subdivisions 175**
48.1 Scope: why this chapter, and what it adds . . . 175
48.2 The diamond as tessellation unit . . . 175
48.3 The dual hexagon hierarchy, and where it diverges from GBT . . . 176
48.4 The kite: a common unit across diamonds, hexagons, and triangles . . . 176
48.5 An arithmetic check of the stated aperture-4 composition . . . 177
48.6 An independent analytic reconstruction of the vertex-centred dual hexagon . . . 177
48.7 Closing the edge-midpoint case at the next finer resolution . . . 178

**49 PYXIS: An Aperture-3 Hexagonal Indexing Scheme, Read from Its Patent 180**
49.1 Scope, and a source substitution disclosed up front . . . 180

49.2 The indexing rule: two cases, stated as a recursion . . . . . 180
49.3 The two addition matrices, transcribed and structurally checked . . . . . 181
49.4 The icosahedral extension, read but not re-derived . . . . . 182

**50 The Ring GBT Is a p-Adic Integer Ring in Disguise 184**
50.1 Scope, and what was actually fetched . . . . . 184
50.2 The construction: GBT as a quotient ring, addresses as an inverse limit . . . . . 184
50.3 The proof's load-bearing steps, read but not re-derived . . . . . 185
50.4 What is checked here: the arithmetic hypothesis, and a cross-check against Ch. 18 185
50.5 Concluding remarks left open by the paper itself . . . . . 186

**51 A Sixth Road, Read Last but Earliest: The Icosahedral Grid of Numerical Weather Prediction 188**
51.1 Scope: a precursor found after the synthesis that should have included it . . . . 188
51.2 The construction, and what it has in common with this volume's other precursors 189
51.2.1 An independent check of the paper's own reported grid size . . . . . 189
51.3 A motivation different again from the other five . . . . . 190
51.4 Revisiting Part One's synthesis . . . . . 190

**52 A Contemporaneous, Independent Sixth Road: Williamson's Spherical Geodesic Grid 192**
52.1 Scope: the companion 1968 paper, read in full via its author's own thesis . . . . 192
52.2 The construction: a geodesic grid, but not the same triangulation as Sadourny's 193
52.2.1 An independent check of the thesis's own reported grid sizes . . . . . 193
52.3 A motivation stated independently, in the same terms . . . . . 194

**53 From One Sphere to Any Polyhedron: Sadourny's 1969 and 1972 Sequels 196**
53.1 Scope: what the 1968 paper's own open ends led to . . . . . 196
53.2 Sadourny & Morel (1969): hexagons on a plane, and the full primitive equations 196
53.3 Sadourny (1972): from the icosahedron to any regular polyhedron . . . . . 197
53.3.1 An independent structural check: the cube as a regular-polyhedron grid 197

**54 Not a Missing Link but a Published One: How the DGGS Field's Own Founders Cited the NWP Grids 199**
54.1 Scope: the question ch:sadourny-icosahedral-nwp left open . . . . . 199
54.2 Sahr, White & Kimerling (2003): a single taxonomy spanning both traditions . 199
54.2.1 A precise, citable resolution of Chapter 52's open geometric question . . 200
54.3 An unresolved implication for Chapter 74's own synthesis . . . . . 201

**55 A Stated Purpose After All: Revisiting White (2000) for Motivation and an Unlooked-For Citation Trail 202**

55.1 Scope: the question ch:global-frame-synthesis's todobox left open . . . . . 202

55.2 White's own stated purpose . . . . . 202

55.3 An unlooked-for citation trail into the NWP tradition . . . . . 203

55.4 Consequence: a further, narrower amendment to Chapter 74 . . . . . 204

**56 Closing (Partly) the Citation-Trail Question: Does Thuburn (1997) or Heikes and Randall (1995a,b) Cite Sadourny (1968) or Williamson (1968) Directly? 206**

56.1 Scope: the first item left open by Chapter 55's todobox . . . . . 206

56.2 Thuburn (1997): resolved . . . . . 206

56.3 Heikes and Randall (1995a,b): not resolved . . . . . 207

**57 Patents on the Sphere: Cartography as Intellectual Property 208**

57.1 Scope: the cost of distortion as a claimed invention . . . . . 208

57.2 Fuller's 1946 patent: uniform peripheral scale as the claimed advance . . . . . 208

57.3 A century of projection and DGGS patents, compared . . . . . 209

57.4 What a projection patent can and cannot claim . . . . . 210

57.5 Closing: an intellectual-property genealogy alongside the mathematical one . . 210

**58 Extending EVT Boundary Laws to the Discrete Case 212**

58.1 What kind of boundary law is needed here . . . . . 212

58.2 The exact boundary law . . . . . 212

58.3 The uniform (bulk) rate, and why it differs . . . . . 213

58.4 Numerical verification . . . . . 213

58.5 Scope and open extensions . . . . . 214

**59 The Dual Scheme, and an Impossibility Theorem 215**

59.1 Why a dual scheme . . . . . 215

59.2 The equal-area scheme and its own boundary law . . . . . 215

59.3 An impossibility theorem . . . . . 216

59.4 A quantitative floor: the universal density-spread theorem . . . . . 217

59.4.1 The two-ring case . . . . . 218

59.4.2 Reduction of every $K$ to the two-ring case . . . . . 218

59.4.3 Establishing the floor of the two-ring function . . . . . 219

59.4.4 Where the natural schemes sit . . . . . 221

**60 Beyond the Equiangular Graticule: Boundary Rates Under General Ring Placement 222**

60.1 The question left open in Chapter 2 . . . 222
60.2 A one-parameter family of ring placements . . . 222
60.3 Exact closed form for the boundary ratio . . . 223
60.4 The asymptotic regime and a phase transition at $\beta = 1/2$ . . . 224
60.5 Does the pole remain the extremal ring? . . . 225
60.6 Scope and open extensions . . . 226

**61 Why Real DGGS Don't Obey the Graticule's Boundary Law 227**
61.1 Scope: the connection item, from two todoboxes at once . . . 227
61.2 The graticule shrinks toward the pole; H3 and S2 don't . . . 227
61.3 The self-similarity argument is a sketch, not a proof . . . 228
61.4 The bulk/extreme distinction, once more . . . 228
61.5 What remains open . . . 228

**62 From the Graticule to a Real Grid: Does the Boundary Law Survive Contact with H3? 229**
62.1 Scope: the item ch07 flagged as not yet attempted . . . 229
62.2 What is actually being compared . . . 229
62.3 Results: three resolutions, three ring counts . . . 230
62.3.1 Is the match specific to $\beta = 1$? . . . 230
62.3.2 Why the profile shape itself is not read the same way . . . 230
62.4 An honest candidate explanation, not a proof . . . 231

**63 Three Grids, One Ring Law: Extending the Check to S2 and rHEALPix 232**
63.1 Scope: closing item (iii) of ch:h3-ring-placement-law's todobox . . . 232
63.2 Results . . . 232

**64 Testing a Candidate Mechanism: Does Within-Band Heterogeneity Explain the Cross-Grid Gap? 234**
64.1 Scope: items (iv) and (v) of ch:ring-placement-law-s2-rhealpix's todobox . . . 234
64.2 Method . . . 234
64.3 Results . . . 235
64.4 What this rules in, and what remains open . . . 235

**65 Shape, Not Area: Testing the Polar-Band IPQ Correlation 237**
65.1 Scope: the second candidate mechanism from Chapter 64's todobox . . . 237
65.2 Method . . . 237
65.3 Results . . . 238

65.4 What the correlation does, and does not, explain . . . 238

**66 The Boundary Itself, Not the Cells: A Third Candidate Mechanism 240**

66.1 Scope: the second untested direction from Chapter 64's todobox . . . 240
66.2 Results . . . 240
66.3 What this rules in, and what remains open . . . 241

**67 Mean Was Not Enough: The Band's IPQ Variance 243**

67.1 Scope: the second item of Chapter 65's todobox . . . 243
67.2 Results . . . 243
67.3 What this rules in, and what remains open . . . 244

**68 A Signed Candidate: The Band's IPQ Skewness 245**

68.1 Scope: the first item of Chapter 67's todobox . . . 245
68.2 Results . . . 245
68.3 What this rules in, and what remains open . . . 246

**69 Sub-Cell Resolution: Closing the Literal Aliasing Test 247**

69.1 Scope: the second item of Chapter 66's todobox . . . 247
69.2 Results . . . 248
69.3 What this rules in, and what remains open . . . 249

**70 Isolating the Last Boundary: Is Band 0 an Accessory? 250**

70.1 Scope: the first item of Chapter 66's todobox . . . 250
70.2 Results . . . 250
70.3 What this rules in, and what remains open . . . 252

**71 An Empirical Case Study: Hexagonal Grids and Generalization in Geospatial Deep Learning 253**

71.1 From geometry to a downstream cost . . . 253
71.2 The systematic sampling-bias problem . . . 253
71.3 Why this belongs in the global frame, not the regional case . . . 254

**72 Hexagon Mosaic Maps: From a Sampling Grid to a Display Convention 255**

72.1 Scope: the same hexagon, read as a display surface rather than a sampling frame 255
72.2 Two companion papers, one grid, one journal volume . . . 255
72.3 The hexagon mosaic map and the ray glyph . . . 256
72.4 Why hexagons: the tessellation choice, argued twice . . . 257
72.5 A 2024 descendant: hexagonal tiles, semantic icons, and an explicit MAUP link 257

72.6 Synthesis: the same cell, asked to carry an estimate and its own uncertainty . . 258

**73 Before the Map: Hexagonal Binning as a Non-Geographic Precursor 260**

73.1 Scope: completing a genealogy left incomplete in the previous chapter . . . . . 260

73.2 The 1987 problem: too many points, not too much geography . . . . . 260

73.3 Why hexagons for binning a plane: the same argument as Chapter 11, one dimension down . . . . . 261

73.4 From a 1987 statistical technique to a name in software . . . . . 261

73.5 Synthesis: the technique found geography twice, not once . . . . . 262

**74 Six Roads to the Same Tessellation: Closing Part One 263**

74.1 Scope: the debt this chapter closes . . . . . 263

74.2 Six precursors, six stated starting problems . . . . . 263

74.3 What converges, and what does not . . . . . 263

74.4 Two gaps this synthesis does not paper over . . . . . 266

**II The Regional Case: Low-Distortion Auxiliary Spheres 268**

**75 Kerkovits's Approach: Regional Rather Than Global Optimization 269**

75.1 The rectifying auxiliary sphere . . . . . 269

75.2 Historical context: rectifying and parametric spheres . . . . . 269

75.3 What "regional" costs, structurally . . . . . 270

**76 Reproducing the Rectifying Sphere: An Independent Derivation 271**

76.1 Scope: closing the open item of Chapter 75 . . . . . 271

76.2 Setting: the ellipsoid-to-sphere step of a double mapping . . . . . 271

76.3 Solving for the free parameters . . . . . 272

76.4 Independent numerical verification . . . . . 272

76.4.1 The acceptance band . . . . . 273

76.4.2 The pole-adapted variant and $n$'s range . . . . . 273

76.4.3 Against the traditional spheres . . . . . 274

76.4.4 The angular counterpart: Tissot $\omega$ for all four spheres . . . . . 275

**77 Asymmetric Placement of the Three Standard Parallels 279**

77.1 Scope: the first item of Chapter 76's closing todobox . . . . . 279

77.2 Construction and cross-check . . . . . 279

77.3 Does asymmetry help, hurt, or neither? . . . . . 280

**78 Regional Versus Global: Where Does Local Optimization Win? 282**
78.1 The research question . . . 282
78.2 The cost of a fixed-width band is not uniform in latitude . . . 282
78.3 Scope: what this result does and does not show . . . 283

**79 A Documented National Case: New Zealand and the Price of a Global Auxiliary Sphere 285**
79.1 Why this case, and what it does and does not share with Chapter 75 . . . 285
79.2 Method . . . 286
79.3 Results: two global grids, one regional one . . . 286
79.4 The actual price of the regional sphere . . . 287

**80 Regional Sphere Against Global Grid: A First Matched-Band Reading 289**
80.1 Scope: a first, deliberately partial answer . . . 289
80.2 Why the two numbers are not simply comparable . . . 289
80.3 Method . . . 289
80.4 Results . . . 290
80.5 Which sphere rHEALPix's area axis should actually be read against . . . 291

**81 Extending the Matched-Band Reading to H3 293**
81.1 Scope: closing part of ch:kerkovits-vs-dggs-band's todobox . . . 293
81.2 Method . . . 293
81.3 Results . . . 293

**82 Extending the Matched-Band Reading to DGGRID and OpenEAGGR 295**
82.1 Scope: closing the item left open by Chapter 81 . . . 295
82.2 Method . . . 295
82.3 Results . . . 296

**83 The Matched-Band Reading Against the Asymmetric, Widened Band 298**
83.1 Scope: does a wider band change the ordering? . . . 298
83.2 Method . . . 298
83.3 Results . . . 298

**84 Closing the DGGRID/OpenEAGGR Item for the Asymmetric, Widened Band 300**
84.1 Scope: closing the item left open by Chapter 83 . . . 300
84.2 Method . . . 300
84.3 Results . . . 301

**85 The Missing Axis: Shape Distortion Within the Same Band 303**
85.1 Scope: closing the first item of ch:kerkovits-vs-dggs-band's todobox . . . . . 303
85.2 Method . . . . . 303
85.3 Results . . . . . 304

**86 H3's Missing Axis: Closing the Angular-Distortion Item 306**
86.1 Scope: the last item of a four-chapter chain . . . . . 306
86.2 Method . . . . . 306
86.3 Results . . . . . 307

**87 Does It Matter Where the Band Sits? 309**
87.1 Scope: the last open item, finally tested . . . . . 309
87.2 Method . . . . . 309
87.3 Results: area axis . . . . . 310
87.4 Results: angle axis . . . . . 311
87.5 What this closes, and what it does not . . . . . 312

**88 Where the Angle-Axis Ordering Actually Flips 313**
88.1 Scope: the first of three remaining items . . . . . 313
88.2 Method . . . . . 313
88.3 Results: the crossover is not one point . . . . . 314
88.4 A longitude that lands exactly on a cube edge . . . . . 315
88.5 What this closes, and what it still does not . . . . . 316

**89 Closing the Gap: The Indicatrix, Real and Measured, for DGGRID and OpenEAGGR 317**
89.1 Scope: the gap left open twice . . . . . 317
89.2 Locating a real face center, and a real transect . . . . . 318
89.3 An equal-area check that falls out of the data itself . . . . . 319
89.4 Step-size stability, and the honest precision ceiling . . . . . 319
89.5 OpenEAGGR: the same map, confirmed from its own source . . . . . 320

**90 New Zealand in the Band: Connecting the National Case to the Abstract Comparison 321**
90.1 Scope: the connection Chapter 79 left for later . . . . . 321
90.2 The coincidence: one country, one matched band . . . . . 321
90.3 What lines up, once the band is shared . . . . . 322

**91 A Second Documented National Case: Switzerland's Century-Old Local Frame 323**

91.1 Scope: a second real data point, not a second method . . . 323
91.2 Method . . . 323
91.3 Results: the tightest distortion budget in this volume, against two loose global ones . . . 324
91.4 The same global-scheme contrast, and one difference worth noting . . . 325
91.5 Closing: two countries, one Mercator variant, two different margins . . . 325

**III The Cost of Distortion: Efficiency Against Accuracy 326**

**92 The Equal-Area / Computational-Efficiency Trade-off 327**
92.1 The tension, stated by its own literature . . . 327
92.2 A toy constrained-optimization formalization . . . 327
92.3 Scope and what a real trade-off analysis would need . . . 328

**93 A Real (Non-Toy) Trade-off: Distortion Against Operational Cost on H3 330**
93.1 Scope: items (i) and (ii) of the toy chapter's todobox . . . 330
93.2 The extended table . . . 330
93.3 What this does, and does not, resolve about the toy trade-off . . . 331

**94 Beyond Storage: Timing the Operations Themselves 333**
94.1 Scope: the second open item of the real-tradeoff chapter . . . 333
94.2 The measured table . . . 333
94.3 What remains open . . . 334

**95 The Same Real Trade-off, on S2 335**
95.1 Scope: the first item of Chapter 93's closing todobox . . . 335
95.2 The table . . . 336
95.3 Diagnosing the offset: ruling out date-line-crossing inclusion . . . 337
95.4 What this resolves, and what it does not, about Chapter 93's open item . . . 338

**96 Timing S2's Operations, the Same Way 339**
96.1 Scope: the second open item of the S2 real-tradeoff chapter . . . 339
96.2 The measured table . . . 339
96.3 What this adds to the S2 real trade-off . . . 340

**97 A Third DGGS, and a Structurally Different Trade-off: rHEALPix 342**
97.1 Scope: closing part of Chapter 11's oldest open item . . . 342
97.2 Area distortion: testing the "equal-area" name directly . . . 342

97.3 Shape distortion: the isoperimetric quotient . . . 343
97.4 Storage cost, measured without an assumed fixed-width index . . . 344
97.5 What this resolves, and what remains . . . 345

**98 Timing rHEALPix's Operations: A Third, Distinct Pattern 346**
98.1 Scope: the operational-timing item left open by Chapter 97 . . . 346
98.2 The measured table . . . 347
98.3 What this adds to the rHEALPix real trade-off . . . 348

**99 Two Open Items Closed: The Neighbour-Branch Mechanism and the Coarsest-Resolution IPQ Artifact 349**
99.1 Isolating the neighbour-branch mechanism . . . 349
99.2 Quantifying the resolution-0 IPQ artifact . . . 351

**100 The Spherical Ceiling Checked: A Latitude-Convention Artifact, Not (Only) Spherical Excess 354**
100.1 An exact ceiling, and an apparent violation of it . . . 354
100.2 Tracing the mechanism . . . 355
100.3 Does the same bias appear at finer resolutions? . . . 356

**101 The Latitude-Convention Bias, Checked for Stability in $n$ and Extended to Resolution 3 358**

**102 S2's Discrete Shape Cost: A Cleaner Story Than rHEALPix's 360**
102.1 Scope: the second grid for the three-way comparison . . . 360
102.2 The refinement check: no artifact to find, anywhere . . . 361
102.3 The genuine curve: S2's discrete shape cost . . . 362
102.4 Placing S2 in the three-way comparison . . . 362
102.5 What remains open . . . 363

**103 Local and Finite-Scale Distortion, Compared 364**
103.1 Scope: three measures of the same grid, never placed side by side . . . 364
103.2 rHEALPix: three measures, one qualitative story . . . 364
103.3 Closing the efficiency thread: what query cost does and does not track . . . 365

**104 The Missing Side, Supplied: S2's Ring-Placement Law Across the Same Levels as Its IPQ Curve 367**
104.1 Scope: closing Chapter 103's remaining gap . . . 367
104.2 Results: the full sweep . . . 368

**105DGGRID: The Missing Fourth Real Trade-off, and a Different Cost Model 369**
105.1Scope: the gap Section 114.5 names twice . . . 369
105.2Method: distortion versus storage . . . 369
105.3What the table shows: a sharp drop, then saturation – and an odd/even wrinkle 370
105.4Method: batch address-conversion timing . . . 370
105.5What the timing shows: flat in resolution, linear in batch size . . . 371
105.6What this resolves, and what remains . . . 372

**106HTM: A Fifth Real Trade-off, and the Cheapest Storage Cost Yet 373**
106.1Scope: the gap this volume's own accounting names . . . 373
106.2The table, read as a trade-off . . . 373
106.3A caveat on the storage comparison itself . . . 374

**107Timing HTM's Operations: Descent Cost, and One Open Item 375**
107.1Scope . . . 375
107.2The third operation this chapter does not measure . . . 375
107.3Results . . . 376

**108OpenEAGGR: A Sixth Real Trade-off, Rebuilt From Source 377**
108.1Scope: extending a two-point sample, as Chapter 105 did for DGGRID . . . 377
108.2Results . . . 378

**109Timing OpenEAGGR's Operations, and a Crash Caught Along the Way 379**
109.1Scope . . . 379
109.2A crash caught, not hidden . . . 379
109.3Results . . . 380

**110GBT's Trade-off: When Distortion Cost Is Exactly Zero 381**
110.1Scope: the trade-off collapses to one dimension . . . 381
110.2Results . . . 381

**111Timing GBT's Own Operations, and a Second Bug Caught Along the Way 384**
111.1Scope: what GBT has to time, and what it does not . . . 384
111.2Results . . . 385

**112The Shape of a Carry Chain: Sampling GBT's Addition Cost 387**
112.1Scope: what Chapter 111 left as a flat mean . . . 387
112.2Method . . . 387
112.3Results . . . 387

**113 An Operational Resolution Crosswalk Across Five DGGS, and Why GBT Stays Out of It** **389**

113.1 Scope: what "unifying resolution numbering" can and cannot mean here . . . . 389

113.2 The aperture check: does $D(r)$ actually behave geometrically? . . . . . 390

113.3 The crosswalk table . . . . . 390

113.4 Why GBT is not in the table . . . . . 391

**114 Where This Volume Stands: An Honest Accounting** **393**

114.1 What this chapter is, and is not . . . . . 393

114.2 What stands on a fully independent derivation . . . . . 393

114.3 What rests on cited external sources, and how firmly . . . . . 394

114.4 What is toy, and clearly labelled as such . . . . . 395

114.5 What has failed, or not yet been attempted, by extrapolation . . . . . 395

114.6 Honest status of the volume as a whole . . . . . 398

**References** **399**

# Preface and Status of This Volume

Monographs 1–4 (collected under the title *No Map Is Best*) treated cartographic distortion as a property of a *continuous* map: a function sending points of the sphere to points of the plane. The three lines of attack developed there — the information-theoretic construction (IDI), the metric-embedding and impossibility apparatus, and the extreme value theory (EVT) boundary laws for latitude bands — all share this continuous frame.

This fifth monograph opens a different, but not unrelated, front: what happens when the reference surface is no longer mapped by a continuous function but tessellated directly into discrete cells (*Discrete Global Grid Systems*, DGGS)? The distortion problem does not disappear in this move; it is reformulated. The thread running through the whole volume, stated in full in Chapter 92, is that the same question — how much does it cost to represent something continuous discretely? — is posed at three different scales: the global scale (Part I), the regional scale (Part II), and the trade-off between accuracy and computational efficiency (Part III).

As of this revision the volume comprises 112 chapters across these three parts; Chapter 114 gives the full, itemized account of what is proved outright, what rests on external sources and how firmly, and what remains open, and should be read as this preface's companion rather than this preface repeating it in summary.

**Methodological Note**

At the date of this draft, this volume is an open construction site and not yet as complete as Monographs 1–4. Every quantitative claim must trace back to a source cited in Section 114.6 or to a computation executed and verifiable within the volume itself (scripts in `code/`, results in `data/`). Sections not yet supported by a primary source or an executed computation are flagged with a "To Complete / Verify" box and must not be read as settled results. Unlike Monographs 1–4, already closed and complete, this monograph is still being built up: the page count reported in this draft is honest and provisional, not a target reached by padding.

# Part I

# The Global Discrete Frame

Chapter 1

# Why Discrete: From the Limits of Continuous Projections to DGGS

## 1.1 The problem inherited from Monographs 1–4

Monographs 1–4 treated distortion as a property of a *continuous* map: a function sending points of the sphere (or ellipsoid) to points of the plane, with distortion measured either locally (the Tissot indicatrix) or on a finite scale (latitude bands, the EVT boundary laws of Chapter 37 of *No Map Is Best*).

*Discrete Global Grid Systems* (DGGS) invert the problem: instead of a continuous function, one starts from a discrete tessellation of the globe into cells, and asks how uniform the area and shape of these cells are. The distortion problem does not vanish; it shifts from "how much does the map deform at a point" to "how much do the cells vary among themselves."

## 1.2 The reference case: Mercator and rectangular cells

The classical didactic argument (Greenland enlarged relative to Africa under the Mercator projection) has a direct discrete analogue: a regular lat/lon grid produces cells whose physical area decreases in proportion to $\cos\phi$ (where $\phi$ is latitude), while the sampling density per unit physical area grows as $1/\cos\phi$ (Author, s). This claim is derived here explicitly, in *exact* form and not merely to first order, because it is the foundation on which the rest of this chapter rests.

**Definition 1.1** ($K$-ring discrete graticule)**.** Divide the quadrant $\phi \in [0, \pi/2]$ (equator–pole) into $K$ equiangular rings of width $\Delta\phi = \pi/(2K)$, ring $i$ being $\phi \in [i\Delta\phi, (i+1)\Delta\phi)$, $i = 0, \dots, K-1$. Within each ring, use a constant longitude step $\Delta\lambda$ (the same for all rings, as in the traditional lat/lon graticule). Let $A_i$ be the physical area (on a sphere of radius $R$) of a single cell of ring $i$.

**Proposition 1.2** (Exact ring-cell area)**.** *Under Definition 1.1,*

$$A_i = R^2\,\Delta\lambda\,\big[\sin((i+1)\Delta\phi) - \sin(i\Delta\phi)\big], \tag{1.1}$$

*and, setting $\rho_i := A_i/A_0$ (area normalized against the equatorial cell),*

$$\rho_i = \frac{\cos\big((i+\tfrac{1}{2})\Delta\phi\big)}{\cos(\Delta\phi/2)}. \tag{1.2}$$

*Proof.* The area of a cell $[\phi_1, \phi_2] \times [\lambda_1, \lambda_1 + \Delta\lambda]$ on a sphere of radius $R$ is $R^2\Delta\lambda \int_{\phi_1}^{\phi_2} \cos\phi\, d\phi = R^2\Delta\lambda(\sin\phi_2 - \sin\phi_1)$, which with $\phi_1 = i\Delta\phi$, $\phi_2 = (i+1)\Delta\phi$ gives Eq. (1.1). For Eq. (1.2), apply Werner's identity $\sin A - \sin B = 2\cos\big(\frac{A+B}{2}\big)\sin\big(\frac{A-B}{2}\big)$ to the numerator with $A = (i+1)\Delta\phi$, $B = i\Delta\phi$: $\sin((i+1)\Delta\phi) - \sin(i\Delta\phi) = 2\cos\big((i+\frac{1}{2})\Delta\phi\big)\sin(\Delta\phi/2)$. The denominator, at $i = 0$, is $\sin\Delta\phi = 2\sin(\Delta\phi/2)\cos(\Delta\phi/2)$. The ratio cancels $2\sin(\Delta\phi/2)$ from numerator and denominator, giving Eq. (1.2). □

*Remark* 1.3 (Continuous limit). As $K \to \infty$ ($\Delta\phi \to 0$) at fixed $i\Delta\phi =: \phi$, $\cos(\Delta\phi/2) \to 1$ and $(i + \frac{1}{2})\Delta\phi \to \phi$, so $\rho_i \to \cos\phi$: Eq. (1.2) is precisely the discrete version, exact at every finite $K$, of the $\cos\phi$ law already used in Monograph 1 as a didactic argument and cited here from Author (s). It is not an approximation of the continuous law applied post-hoc to finite cells: it is a self-standing proposition about the area of finite cells, of which the $\cos\phi$ law is the $K \to \infty$ limit.

Table 1.1 checks Eq. (1.2) against direct computation Eq. (1.1) (difference of sines) for six values of $K$ and three rings per value (equator, mid-latitude, adjacent to the pole).

Table 1.1: Verification of Eq. (1.2): $\rho_i$ computed directly from Eq. (1.1) against the closed form, for $K \in \{4, 8, 16, 32\}$ (a further two decades, $K = 64, 128$, are used in Chapter 58). Source: `code/graticule_ring_area_check.py`, `data/graticule_ring_area_results.json`.

| $K$ | ring $i$ | $\rho_i$ (direct) | $\rho_i$ (closed) | $\lvert\Delta\rvert$ |
|---|---|---|---|---|
| 4 | 0 | 1.000000 | 1.000000 | 0 |
| 4 | 2 | 0.566454 | 0.566454 | $1.1 \times 10^{-16}$ |
| 4 | 3 | 0.198912 | 0.198912 | 0 |
| 8 | 0 | 1.000000 | 1.000000 | 0 |
| 8 | 4 | 0.637463 | 0.637463 | $3.3 \times 10^{-16}$ |
| 8 | 7 | 0.098491 | 0.098491 | $5.6 \times 10^{-17}$ |
| 16 | 0 | 1.000000 | 1.000000 | 0 |
| 16 | 8 | 0.672369 | 0.672369 | $1.0 \times 10^{-15}$ |
| 16 | 15 | 0.049127 | 0.049127 | $5.9 \times 10^{-16}$ |
| 32 | 0 | 1.000000 | 1.000000 | 0 |
| 32 | 16 | 0.689748 | 0.689748 | $1.6 \times 10^{-15}$ |
| 32 | 31 | 0.024549 | 0.024549 | $2.3 \times 10^{-16}$ |

The differences remain at the level of double-precision rounding error ($\sim 10^{-15}$), as expected: Eq. (1.2) is an exact trigonometric identity, derived algebraically from Eq. (1.1), not an asymptotic result to be confirmed by simulation.

**Key Result**

The $K$-ring lat/lon graticule has an *exact* cell-area law Eq. (1.2), whose $K \to \infty$ limit is the $\cos\phi$ law cited as a known fact at the opening of this chapter. Section 58.1 in Chapter 58 uses this same formula to derive the behaviour of cells near the pole — the discrete analogue of the boundary EVT laws of *No Map Is Best*, Ch. 24 and Ch. 31/34.

## 1.3 Five open-source implementations compared

Kmoch et al. (2022) compare five open-source DGGS implementations — Uber H3, Google S2, RiskAware OpenEAGGR, rHEALPix (Landcare Research NZ), and DGGRID (Southern Oregon University) — on the basis of normalized cell area and compactness. The authors' conclusion is that the rHEALPix, OpenEAGGR, and DGGRID definitions, all built on an ISEA (Icosahedral Snyder Equal Area) base, offer the strongest guarantee of global-scale area preservation, while H3 shows the smallest shape distortion but, together with S2, the largest cell-area variation (Kmoch et al., 2022).

**To Complete / Verify**

Reproduce (or at minimum tabulate with exact figures drawn from the paper) the normalized-area and compactness values reported by Kmoch et al. (2022), rather than stopping at the qualitative conclusion above. The authors released code and data supplements (Kmoch & Vasilyev, 2022a, 2022b) that should make this reproducible directly rather than merely re-cited; this reproduction is left for future work, consistent with the no-invented-data discipline: no numeric value from that paper is stated here beyond what has already been directly verified.

**Methodological Note**

The full bibliographic record for Kmoch et al. (2022) — *Big Earth Data*, volume 6, issue 3, pages 256–275, DOI 10.1080/20964471.2022.2094926, received 19 February 2022, accepted 15 June 2022, published online 29 July 2022 — has been verified directly against the publisher's abstract page and against the University of Tartu Landscape Geoinformatics group's own publication list, and is not merely carried over from a search snippet.

## 1.4 What Chapters 2–9 do with this frame

Chapter 58 extends the exact ring-area law of Proposition 1.2 to a genuine boundary-law statement in the EVT sense used throughout *No Map Is Best*: not merely that cells shrink toward the pole, but at what *rate*, and how that rate compares to the uniform (bulk) discretization error away from the pole. Chapter 59 then asks whether the equiangular and equal-area conventions can ever coincide, proving they cannot for $K \geq 2$ and establishing a universal floor on how far apart their local densities must remain. Chapter 60 takes up the ring-placement question directly: once the equiangular convention is dropped in favour of a general, parametrized family of placements, the pole's boundary rate is shown to undergo a sharp phase transition rather than a smooth rescaling. Chapter 71 then leaves the purely geometric register and asks whether area/shape non-uniformity has a measurable downstream cost, using a concrete 2026 empirical study of deep-learning generalization on hexagonal versus lat/lon grids. Part II moves from the global to the regional register, and Part III formalizes the efficiency/accuracy trade-off that both registers ultimately share.

Chapter 2

# The Graticule as the First DGGS

## 2.1 Scope: a discrete scheme with no hexagons, triangles, or code

Chapter 1 frames this volume's central move as the passage from continuous projections to discrete, hierarchically tessellated grids, and the chapters that follow trace six independent twentieth- and twenty-first-century roads to that move: H3, S2, rHEALPix, DGGRID, OpenEAGGR, and the older QTM/HTM/GBT/PYXIS/NWP-grid family (Section 1.4 and Chapter 74). All of these share a property this chapter argues is not, in fact, essential to being a discrete global addressing scheme: recursive subdivision of a polyhedron into small, area- or shape-controlled cells. This chapter looks instead at the oldest scheme that assigns every point on Earth a discrete, finite-precision, hierarchically refinable address — the graticule of parallels and meridians, in the specific, coordinate-based form Ptolemy describes and implements in his *Geography* — and asks what is gained, and what is given up, by insisting (as every later chapter's DGGS does) on a tessellation of roughly equal-area or roughly equal-shape cells instead.

**Methodological Note**

This chapter introduces no new source to Section 114.6. It draws entirely on Miyachi et al. (2025), already fetched and cited in Chapter 30 for its bi-Lipschitz reformulation of Tissot's indicatrix, here reused for a different part of the same source: its Section 3–4 biographical and textual account of Marinos of Tyre and Ptolemy, including its direct quotations from Ptolemy's *Geography* (via the Berggren–Jones English translation the source itself cites). No new computation is performed.

## 2.2 Marinos, Ptolemy, and the first documented coordinate grid

Miyachi et al. (2025) reports that Marinos of Tyre (*c.* 70–130 CE), whom they describe as sometimes called the father of mathematical geography, is known to us only through Ptolemy's own *Geography*, which quotes and criticizes him at length. Per Miyachi et al. (2025), Ptolemy's text records that Marinos did two things this chapter treats as jointly constituting a discrete global addressing scheme, prior to and independent of any map drawing: he *accurately noted a large number of coordinates of places on Earth, by their latitude and longitude*, and he used this coordinate data, with corrections, as the input to his own (criticized) method of map construction.

**Key Result**

On Miyachi et al. (2025)'s account, a coordinate — a pair of angular quantities locating a place with respect to a fixed origin meridian and the equator — existed, was recorded for real places, and was used as an addressing and data-interchange format *before* any particular map projection was chosen to draw those places. This is the defining separation this volume's own DGGS chapters draw between a grid's *addressing* function (assigning a dis-

crete, hierarchical code to every location) and its *projection* function (mapping that structure to a plane or a set of polyhedron faces for display or storage), except that Marinos's version of the separation predates the DGGS literature's version of it by roughly nineteen centuries, and predates any tessellation into cells of controlled area or shape by the same margin.

Miyachi et al. (2025) adds a philological detail worth recording on its own: citing Tannery, the source notes that the technical term "latitude" was first introduced by Ptolemy himself — that is, not merely the coordinate concept but its standard Western name is a first- or second-century cartographic coinage, not a later retrofit onto older Greek geography.

## 2.3 The grid as Ptolemy specifies it

Miyachi et al. (2025) quotes Ptolemy's own closing words to Book II of the *Almagest*, where, having tabulated stellar coordinates, Ptolemy turns to terrestrial ones:

> *We shall indicate the place of each city by how many degrees, counted on the meridian, it is far from the equator; and in degrees, counted on the equator, the distance to the East or to the west of each meridian, to the one passing by Alexandria…*

This is, in the vocabulary this volume uses for modern DGGS, a complete specification of an addressing scheme: a fixed prime meridian (Alexandria, in Ptolemy's case, playing the same referential role WGS84's Greenwich meridian plays in Chapter 27), two angular coordinates measured in a named unit (degrees), and an explicit statement that every named place receives such a pair. Per Miyachi et al. (2025), Ptolemy's *Geography* goes further and specifies, in Book I, exactly how these coordinates are to be turned into a drawn map: meridians as straight lines through a common point, parallels as concentric circular arcs about that same point, with two named reference parallels (through the island of Thule and the equator, and through the island of Rhodes) held to true scale and every other parallel's spacing fixed relative to those two. The source quotes Ptolemy directly on this construction, criticizing Marinos's cruder all-straight-line version in the same passage.

**To Complete / Verify**

Miyachi et al. (2025) does not discuss, and this chapter therefore does not claim to establish, exactly when the underlying *concept* of a systematic degree-based coordinate grid first appears — only when the first coordinate table and explicit construction rule that survive to us, Ptolemy's, were written down. The source mentions in passing that Hipparchus and Eratosthenes are counted among "the first great Greek geographers" who used astronomical data to fix coordinates, placing systematic latitude/longitude measurement at least two to three centuries before Ptolemy, but gives no further technical detail on either figure's own scheme. A future revision should locate and read a primary or close-secondary source on Eratosthenes's and Hipparchus's own grid-construction methods before extending this chapter's claims earlier than Marinos and Ptolemy.

## 2.4 What the graticule gets right, and what every later DGGS chapter is about

Read against Chapter 1's framing, the Ptolemaic graticule already satisfies several of the properties this volume's later DGGS chapters treat as design goals: it is *global* (every point on the inhabited world receives a coordinate, not just a locally surveyed region); it is *hierarchical* in the loose sense that a coordinate given to whole degrees can be refined to fractions of a degree without changing the addressing scheme itself; and it is *discrete* in the operationally relevant sense that any finite-precision statement of a location — "so many degrees from the equator, so many from Alexandria" — is a code, not a continuously varying real-valued position, the moment it is written down to a stated number of digits.

**Key Result**

What the graticule does *not* do, and what every chapter from Chapter 11 onward in this volume's own Part I is, in one sense, an extended argument for doing instead, is control the *shape or area of the region a coordinate pair addresses*. A one-degree-by-one-degree graticule cell near the equator and a one-degree-by-one-degree graticule cell near a pole cover wildly different surface areas and wildly different shapes (a near-square region shrinking to a near-triangular sliver), a fact Chapter 58's boundary-rate analysis quantifies precisely for the equiangular case specifically because the equiangular case is, at bottom, nothing other than the Ptolemaic graticule read as a tessellation rather than as a coordinate system. Every hexagonal, hierarchical, polyhedron-based DGGS this volume examines can be described, without exaggeration, as a nineteen-centuries-later attempt to keep the graticule's global, hierarchical, discrete addressing property while removing its area and shape nonuniformity — the precise trade this volume's title announces as "the cost of distortion."

## 2.5 Closing: the graticule is not a straw man

It would be a mistake to read this chapter as recasting the graticule merely as the primitive predecessor every later chapter improves on. Miyachi et al. (2025) is explicit that the Ptolemaic construction — straight meridians, circular parallels, two true-to-scale reference parallels — was not a naive default but a considered response to a specific complaint Ptolemy raises against Marinos's simpler all-straight-line grid: that a globe viewed from the front shows its meridians converging, and a flat map that ignores this convergence misrepresents the sphere's own visual and metrical structure. In the vocabulary Chapter 30 and Chapter 34 later develop for this volume, Ptolemy is already trying to manage a distortion trade-off, roughly two thousand years before Tissot supplies a pointwise instrument for measuring the trade-off he was managing by eye and by the two true-parallel constraint alone. The graticule is not the alternative to a distortion-aware discrete scheme; read carefully, via Miyachi et al. (2025)'s account of what Ptolemy actually wrote, it is the first one.

Chapter 3

# Transverse Mercator Zones as a Historical Precedent for Regional Distortion Optimization

## 3.1 Scope: zoning, not tessellation

Chapter 2 argued that the Ptolemaic graticule already separates a discrete, hierarchical *addressing* scheme from any particular *projection* of that scheme onto a plane, and that every later DGGS in this volume's Part I can be read as an attempt to keep that separation while replacing the graticule's own non-uniform cell shapes and areas with roughly equal-area or equal-shape cells. This chapter takes up a different, and in one sense more direct, nineteenth- and twentieth-century precedent for this volume's central concern: not a change of cell *shape*, but a change of cell *scope*. Rather than tessellating the sphere into small controlled-shape cells, the transverse Mercator zone systems this chapter examines — the German Gauss-Krüger tradition and its American Universal Transverse Mercator (UTM) successor — keep a single continuous conformal projection but restrict each instance of it to a narrow longitudinal band, repeating the same projection formula, with a fresh central meridian, in every band. Chapter 78 and Chapter 75 already treat this volume's own regional-versus-global distinction for DGGS grids; this chapter's claim is that the zoned transverse Mercator is a direct cartographic ancestor of that same distinction, predating Kerkovits's regional DGGS optimization (Chapter 75) by roughly a century and the DGGS literature generally by longer still.

**Methodological Note**

This chapter introduces two new sources to Section 114.6. The first is a substantial front-matter excerpt (pp. 1–24 of the 1987 printing) of Snyder (1987), already cited elsewhere in this volume (Chapter 27) for the auxiliary-sphere concept but not previously fetched in full; the newly read pages cover the history of USGS projection choice, the ellipsoid/datum history table, and the U.S. Coast and Geodetic Survey's own 1930s State Plane Coordinate System episode. The second is OGP/POSC (1998–2002), the EPSG/POSC *Guidance Note Number 7*, fetched and read in full for this chapter, which supplies the direct historical statement this chapter is built around. A third source, Wikipedia contributors (2026), is used only, and flagged as tertiary, for the Gauss/Krüger authorship-and-date attribution, cross-checked against Esri's ArcGIS Pro documentation for the same two names and years. No computation is performed; this chapter is entirely a historical-sourcing exercise, in the manner of Chapter 2.

## 3.2 From Lambert and Gauss to Krüger: the ellipsoidal transverse Mercator

The transverse Mercator family starts from the ordinary (spherical) Mercator projection rotated ninety degrees, so that the cylinder of projection is tangent along a chosen meridian rather than along the equator. Per Wikipedia contributors (2026) — cross-checked against Esri's independent statement of the same attribution — the spherical form is due to Johann Heinrich Lambert in 1772; the ellipsoidal correction needed for large-scale, high-accuracy mapping was first worked out by Carl Friedrich Gauss in 1822, and given the further analysis by which the projection is now most often known — Gauss-Krüger — by Johann Heinrich Louis Krüger in 1912. Snyder (1987) independently corroborates, in the material fetched for this chapter, that ellipsoidal geodesy generally was a nineteenth-century preoccupation: the same decades that produced the Gauss ellipsoidal correction also produced the Bessel 1841, Clarke 1866, and Clarke 1880 reference ellipsoids (Snyder, 1987's Table 1, read directly), the raw material any transverse Mercator implementation needs before it can compute a single coordinate.

**To Complete / Verify**

**Substantially resolved.** Neither source originally fetched for this chapter gave a primary citation to Gauss's own 1822 paper or to Krüger's 1912 monograph; both dates rested on tertiary attribution (Wikipedia contributors, 2026, cross-checked only against a second tertiary source, Esri's documentation). This revision locates both: Gauss (1825) is now cited by its full German title, with its composition-in-1822/publication-in-1825 history corroborated by an academic history-of-mathematics source (Voss, 2016) that quotes Gauss's own collected works (*Werke*, Vol. 8, p. 371) rather than an encyclopedia summary; and Krüger (1912)'s publication record is now confirmed via three independent institutional sources – the GFZ Potsdam repository (successor to the original publishing institute), a HathiTrust library catalogue record, and an antiquarian physical description – agreeing on publisher, series, volume number, and extent. Neither monograph's full mathematical text was read in this pass; only their publication records are now primary- rather than tertiary-sourced.

## 3.3 The zone as a distortion-control device

A single transverse Mercator projection, run without limit east and west of its central meridian, suffers the same fate as the ordinary Mercator run without limit toward the poles: scale distortion grows without bound away from the line of true scale. The transverse Mercator's line of true scale is the central meridian itself (or, with a scale-factor reduction, two lines symmetric about it), so distortion grows with longitudinal distance from that one meridian rather than with latitude. The zone systems this chapter is about respond to this by never using one transverse Mercator projection for the whole Earth: instead, the ellipsoid is partitioned into a sequence of narrow longitudinal bands, each just wide enough that distortion stays within a tolerable bound, and each band gets its own instance of *the same projection formula* with its own central meridian.

**Key Result**

OGP/POSC (1998–2002) (its §1.4.6.2, on what it calls the "Transverse Mercator Zoned Grid System") is explicit that this zoning procedure was adopted by German mapping in the

1930s through the Gauss-Krüger systems, and was only later taken up by American military mapping through the Universal Transverse Mercator grid. Structurally, this is exactly this volume's regional-versus-global distinction (Chapter 78), decades before that distinction is drawn for DGGS: rather than accept one global scheme's worst-case distortion everywhere, or design a bespoke projection for every country, the zoned transverse Mercator keeps one universal, mechanically identical rule (the same formula, the same standard parameters, applied verbatim in every zone) and lets *which narrow region* that rule is centered on vary. Kerkovits's DGGS optimization (Chapter 75) is, in this specific sense, a twenty-first-century analogue run over a fundamentally different kind of tessellation — discrete cells rather than continuous longitudinal bands — but aimed at the identical goal: hold a single distortion budget everywhere by localizing the region over which any one instance of the scheme is asked to perform well.

The specific mechanics, per OGP/POSC (1998–2002), are worth recording because they show the zoning idea was engineered, not merely observed after the fact. Four of the transverse Mercator's five defining parameters — latitude of natural origin, scale factor at the natural origin, false easting, and false northing — are held identical across every zone of a given zoned system; only the longitude of the natural origin (the central meridian) changes from zone to zone, and it changes by a fixed, regular increment (the zone width) from a stated initial longitude. For the specific case of UTM, OGP/POSC (1998–2002)'s Table 2 (read directly) gives sixty zones worldwide, each six degrees wide, with a central-meridian scale factor fixed at $0.9996$ in every zone and a false easting of $500{,}000$ m at every zone's own central meridian — the last detail meaning, per OGP/POSC (1998–2002), that every zone's easting values are formally identical to every other zone's, and a coordinate is disambiguated only by an explicit, separately carried zone number prefixed to the easting.

## 3.4 An American instance: the State Plane Coordinate System

Snyder (1987)'s fetched front matter supplies a directly relevant domestic episode, and one this chapter treats as a second, independent instance of exactly the same zoning logic rather than as mere background. In the 1930s, surveyors at the U.S. Coast and Geodetic Survey established the State Plane Coordinate System (SPCS), explicitly influenced, per Snyder (1987), by military and civilian mapping practice already established in Europe. The SPCS assigns each U.S. state one or more zones, each using whichever of two projections suits the state's own predominant extent: the Lambert Conformal Conic for states elongated east-to-west, and the transverse Mercator for states elongated north-to-south — with the choice of projection family itself already a form of regional customization, on top of the zone-by-zone re-centering each family separately provides. That the same decade sees both the German Gauss-Krüger rollout and the American SPCS rollout, with Snyder (1987) explicitly crediting European precedent for the latter, is consistent with OGP/POSC (1998–2002)'s ordering (Germany first, American military mapping later), though neither source fetched for this chapter states a direct transmission channel between the two 1930s episodes, and this chapter does not claim one beyond what Snyder (1987)'s own general acknowledgment of European influence supports.

## 3.5 How far a zone travels: a partial map of adoption

OGP/POSC (1998–2002)'s Table 2 also records, for the Gauss-Krüger method specifically, that it saw use in the former USSR, the former Yugoslavia, Germany, parts of South America, and

China — a spread this chapter reads as evidence that the zoning device, once engineered, was portable across national mapping programs with unrelated ellipsoids and datums, exactly as a DGGS's addressing layer is designed to be portable across applications. UTM's own listed area of use, by contrast, is simply worldwide coverage from 84°N to 80°S, the zoned device's logical endpoint: a single, uniform, globally defined zone grid rather than a patchwork of separately adopted national instances of the same idea.

**To Complete / Verify**

**Narrowed, not fully closed.** Neither source originally fetched for this chapter gave the exact year Germany first mandated Gauss-Krüger zoning, nor the specific zone width of that first rollout; OGP/POSC (1998–2002) states only "the 1930s" without a primary German source. A German-language geodesy-history source, Vermessungsgeschichte.de (2003), now supplies a specific event and date – not merely a decade – citing *Zeitschrift für Vermessungswesen* Sonderheft 23 (1985): the *Beirat für das Vermessungswesen*'s second session, in **May 1923**, recommended Gauß-Krüger coordinates in **3-degree**-wide meridian zones with integer central meridians (3°, 6°, ...), a full decade earlier than OGP/POSC (1998–2002)'s figure. This is a recommendation by an advisory board, not necessarily the moment of legal or regulatory mandate OGP/POSC (1998–2002) may have meant by "the 1930s," so the two figures may both be correct, for different milestones (recommendation versus full legal adoption); this chapter does not adjudicate which milestone either source intends. A second German-language source consulted in the same pass reports a 4-degree zone width with 1-degree overlap for the 1923 event, against the 3-degree figure above; this discrepancy is disclosed rather than resolved, since the underlying 1985 ZfV special issue itself was not fetched directly in either case. The 1985 ZfV Sonderheft 23 remains the next, more primary source to locate.

**Update: substantially resolved.** Vermessungsgeschichte.de (2003) itself turns out to reproduce the Beirat's 1923 recommendation verbatim, quoted directly from *Zeitschrift für Vermessungswesen* Sonderheft 23 (August 1985), p. 87, specifying only integer-degree central meridians (3°, 6°, ...) with no overlap mentioned. This is a direct quotation of the primary 1985 special issue, not a paraphrase, so the 3-degree figure is now supported by a verbatim rendering of the underlying primary document, while the competing 4-degree/1-degree-overlap claim is not shown to rest on anything more primary than the second source's own paraphrase. This does not positively rule out the second source's figure as describing some other, later or regional, variant of the rollout, and the 1985 issue itself was still not fetched directly by this volume.

**Update: source identified, not yet fetched.** The article behind *Zeitschrift für Vermessungswesen* Sonderheft 23 (1985) is now identified: Hunger (1985), "Geschichte des Deutschen Vereins für Vermessungswesen (DVW), Teil II: 1918/19– 1945/50," *ZfV* Sonderheft 23/1985, 110. Jg., pp. 75–136 — independently confirmed by its citation, with the same pages, in a 2025 *zfv* anniversary retrospective. Page 87, the page Vermessungsgeschichte.de (2003) quotes for the 3-degree figure, falls inside this article's own 75–136 span, which is consistent with (though it does not by itself prove) the quotation being drawn from this specific piece rather than elsewhere in the Sonderheft. A bibliography entry for Hunger (1985) has now been added, with author, title, and page span independently corroborated by three further sources found in this pass (see references.tex for detail); the Sonderheft's own text is still not fetched directly by this volume. Separately, current German cadastral practice — Gauß- Krüger zones are still administered today on a 3-degree width, per routine practitioner documentation outside the historical literature — is consistent with the 3-degree figure surviving into present-day use, against the competing 4-degree/1-degree-overlap report; this is corroboration of *outcome*, not of the 1923 *event* itself, since continuous

3-degree practice since 1923 is assumed here rather than independently traced. The Sonderheft's own text, pp. 75–136, is still not fetched directly by this volume, and this is where the item remains: narrowed twice now, closed neither time.

## 3.6 Closing: the same trade, a different tessellation

Read together with Chapter 2, this chapter completes a two-part historical point this volume's later DGGS chapters can be read against. The graticule chapter showed that a global, hierarchical, discrete addressing scheme — Ptolemy's coordinate pairs — existed nineteen centuries before any DGGS tessellated the sphere into cells of controlled shape or area. This chapter shows that the complementary move — restricting a continuous, otherwise-unbounded projection to a bounded region small enough to keep its distortion under control, and repeating that same restricted projection systematically across the whole globe — is itself a full century older than this volume's own DGGS-era treatment of the regional-versus-global choice (Chapters 75 and 78). Neither the Ptolemaic graticule nor the zoned transverse Mercator is a straw man predecessor solved once and for all by later work; each is, on its own terms, an engineered response to the identical underlying fact this volume's title calls the cost of distortion — a fact no scheme, discrete or continuous, tessellated or zoned, has ever been able to make disappear rather than merely relocate.

Chapter 4

# The Military Grid Reference System: Alphanumeric Indexing on Top of a Continuous Projection

## 4.1 Scope: from zoning to addressing

Chapter 3 treated the zoned transverse Mercator — Gauss-Krüger and its UTM successor — as a change of projection *scope*: the same conformal formula, repeated in narrow longitudinal bands rather than applied once to the whole globe. This chapter follows the same nineteen-forties institutional lineage one step further, to a system built directly *on top of* UTM (and its polar complement, the Universal Polar Stereographic, UPS) that this volume has not yet discussed: the Military Grid Reference System (MGRS), formally adopted for U.S. Department of the Army use on 29 December 1949. Where UTM supplies a continuous $(easting, northing)$ coordinate pair inside each zone, MGRS overlays a second, explicitly discrete structure — a hierarchical alphanumeric address, not a numeric coordinate — that identifies nested square cells at fixed metric resolutions (100,000 m, 10,000 m, down to 1 m) without ever requiring the numeric UTM coordinate itself to be transmitted in full. This is, in the terms this volume's Part I uses throughout, a genuine *indexing* scheme — a way of naming discrete cells of a tessellation so that nearby cells have related names and a partial address still disambiguates a region — built by military cartographers a decade before Dutton's QTM (Chapter 37) and more than sixty years before H3, S2, or rHEALPix, on a substrate that is emphatically not equal-area or equal-shape: an ordinary square grid laid over a conformal projection.

**Methodological Note**

This chapter introduces four new sources to Section 114.6, all fetched and read directly in this pass. The first is a set of scanned 1940s archival documents — the December 1946 Army Map Service memorandum to the Chief of Engineers, the November 1945 letter from Brigadier Martin Hotine (Directorate of Military Survey, War Office) to Colonel A. G. Matthews, the undated *Study and Discussion of Military Grids* report, the 27 February 1948 Joint Mapping Photography Committee proposal, and geodesist John W. Hager's 1994 retrospective summary letter — transcribed and hosted together by MapTools.com (MapTools.com, 2026), credited there to a 1994 Pentagon records request. The second and third are Wikipedia contributors (2026) and a companion tertiary source, Vanderhaeghe (2023), both used only for present-day MGRS specification details (the AA/AL lettering schemes, polar UPS conventions, and the point that MGRS is formally a geocoding standard maintained by the NGA Office of Geomatics, distinct from the UTM projection it sits on) not covered by the 1940s archival material, since MGRS's structure has been revised since 1949 and the archival documents describe only the system as originally proposed. No computation is performed; this chapter, like Chapter 3 and Chapter 2, is entirely a historical-sourcing exercise.

## 4.2 The problem MGRS was built to solve: too many incompatible grids

The archival *Study and Discussion of Military Grids*, reproduced in full by MapTools.com (2026), opens by cataloguing the problem rather than the solution: by the end of the Second World War, Allied forces were operating under what the report calls an "undesirable heterogeneous system" of roughly eighty incompatible regional and national grids — the U.S. polyconic military grid, a separate Lambert conformal harbor-defense grid, some twenty U.S. state plane coordinate systems, and a patchwork of British, French, German, and other national grids inherited from prewar mapping. The report is explicit that this was not a merely administrative inconvenience but an operational hazard: it recounts that incompatible grids used simultaneously by British coast artillery, the Navy, and the Air Force in the same coastal sector contributed to "intolerable confusion" during the Battle of Britain, and states as a matter of institutional folk wisdom, only half in jest, that "battles are always fought on grid junctions" (MapTools.com, 2026). Brigadier Hotine's November 1945 letter, addressed to the American side of the same discussion, independently confirms the urgency from the British perspective and records that Britain was, at that date, still weighing a graticule-based "mesh" alternative to any plane grid at all before ultimately concurring with the American six-degree transverse Mercator proposal (MapTools.com, 2026).

The same 1946 Army Map Service study works through a comparison of five candidate grid projections — polyconic, Cassini-Soldner, Bonne, stereographic, and Lambert — against the eventual choice, the transverse Mercator, using a tabulated relative-error metric at a fixed $45{,}000$-yard range from the zone center (MapTools.com, 2026). The comparison is framed entirely in artillery terms, not cartographic ones: the report's stated requirement is that grid-induced range and deflection error stay below the probable error of the largest guns then in service, and its verdict for the transverse Mercator is explicit on exactly the distortion property this volume tracks throughout — that the grid "is conformal and is immediately applicable without plottable error, to the majority of the map projections commonly encountered on the native maps of the world," and that it "reduces inaccuracies to a point where they are compatible with the accuracies required by all modern artillery weapons" (MapTools.com, 2026). The document's own proposed specification — six-degree zones, $0.9996$ central-meridian scale factor, $500{,}000$ m false easting — is, parameter for parameter, the UTM specification this volume already cites in Chapter 3.

## 4.3 From numeric coordinates to alphanumeric addresses

The Joint Mapping Photography Committee's 27 February 1948 proposal, also reproduced by MapTools.com (2026), is the document that adds the genuinely new structure this chapter is about: it specifies dividing the $80°$N–$80°$S band of the world into $8° \times 6°$ rectangles, each identified by a UTM zone number (already numeric, 1–60) followed by a *letter*, not a number, for the $8°$ latitude row — and then subdividing each such rectangle into $100{,}000$-m squares, each identified by a two-letter column-and-row code drawn from a restricted alphabet that deliberately omits the letters I and O to avoid confusion with the digits 1 and 0 (MapTools.com, 2026). The full reference for a point is then this two-part alphanumeric prefix followed by only as many numeric easting/northing digits as the required precision demands — the same reference, truncated rather than rounded, can be read at $100{,}000$ m, $10{,}000$ m, $1{,}000$ m, or finer resolution simply by keeping or dropping trailing digit pairs (Wikipedia contributors, 2026). Wikipedia contributors (2026) documents that this design choice — an alphanumeric label

naming a *cell* rather than a numeric pair naming a *point* — has persisted essentially unchanged into the present-day specification, with the one first-row lettering convention (whether the row adjacent to the equator starts at A or at F, and analogous variants for older datums) formalized as the modern system's "AA" and "AL" schemes.

**Key Result**

Read against this volume's Part I vocabulary, the 1948 MGRS proposal is doing something that later DGGS literature treats as a distinct, often celebrated, design problem: separating a *continuous* coordinate (the UTM easting and northing, which can be interpolated to arbitrary precision) from a *discrete, hierarchical, named* address for the cell containing it, with the crucial property that a truncated address still identifies a valid, larger enclosing cell rather than becoming meaningless. This is precisely the addressing property Chapter 2 attributes to the Ptolemaic graticule and that every aperture-based DGGS in this volume's later chapters (Chapter 25) re-derives for hexagonal or triangular cells. MGRS reaches the same structural idea — discrete, truncatable, hierarchical addressing, layered on top of a continuous projected coordinate rather than replacing it — roughly a decade before Dutton's QTM (Chapter 37) is usually cited as the starting point of the DGGS addressing literature, and does so on an ordinary square grid over a conformal cylindrical projection rather than on any equal-area polyhedral substrate.

## 4.4 What MGRS does not solve: junctions and the limits of zoning

The same archival sources are candid about what the scheme does not fix. MapTools.com (2026)'s Study and Discussion notes that grid-zone junctions remain unavoidable trouble spots, restating the same complaint Hotine's letter raised about every plane-grid system considered; the 1948 proposal's own choice of $8°$-tall latitude bands, rather than a rounder number, is presented as a partial mitigation, not an elimination, of the problem. A full century later, Wikipedia contributors (2026) documents that this is still an open practical issue: the boundaries of MGRS's lettered latitude bands are parallels of latitude, while the boundaries of its $100{,}000$-meter grid squares follow the underlying UTM easting/northing grid, and the two boundary systems do not coincide, so a single $100{,}000$-meter square can straddle two different latitude-band letters and be validly labeled either way; Wikipedia contributors (2026) traces a documented 1999 software fix, prompted by fighter pilots who found that truncating an MGRS string they had received could inadvertently place them on the wrong side of such a boundary, to exactly this residual seam. This is the same genus of difficulty — discrete cells stitched onto a grid whose boundaries do not track any single one of the cell system's own seams — that Chapter 32 and Chapter 60 document for the cube-sphere DGGS family this volume treats at length; MGRS's zippers are simply an earlier, cruder instance of the same structural cost.

**To Complete / Verify**

**Resolved.** This chapter's numeric detail on the modern MGRS specification had rested entirely on Wikipedia contributors (2026), a tertiary source, cross-checked only against a second tertiary summary (Vanderhaeghe, 2023); neither the current NGA Office of Geomatics page nor a primary standard document had been fetched directly. This revision fetches NGA Office of Geomatics (2026), the Office of Geomatics's own current coordinate-systems page, directly (last modified 11 May 26 per its own footer) and confirms every post-1949 numeric detail this chapter uses against it independently: the $6°\times8°$ grid zone designator, the twenty latitude bands lettered C through X omitting I and O (band X

alone spans 12° rather than 8°), the 100,000-meter square lettering, and the truncation-not-rounding convention. It also identifies the current formal specification by its present NGA standard number, **NGA.STND.0037, "Universal Grids and Grid Reference Systems,"** which supersedes the older DMA Technical Manual 8358.1 this chapter's tertiary sources had named; that standard document itself (hosted at NGA's restricted `nsgreg.nga.mil` registry) was not independently fetched, so its exact wording is still tertiary-sourced via NGA Office of Geomatics (2026)'s summary of it, but the chapter's numeric claims are now checked against NGA's own current page rather than against Wikipedia alone.

Chapter 5

# Discrete Addressing Without a Polyhedron: Geohash and Open Location Code

## 5.1 Scope: the seventh scheme, and the one that skips a step

Every discrete indexing scheme this volume has examined so far — MGRS (Chapter 4), H3, S2, rHEALPix, DGGRID/ISEA3H, HTM, OpenEAGGR, and GBT/MGBT — shares one structural commitment even where it shares nothing else: each first fixes a polyhedron (a cube, an icosahedron, an octahedron) or an auxiliary sphere, projects the graticule onto or through it, and only then subdivides the result into addressable cells. Chapter 2 showed that this volume's own opening precedent, the bare latitude/longitude graticule, lacked exactly that intermediate step, and paid for the omission in extreme polar-cell distortion. This chapter examines two widely deployed, still-active discrete indexing systems — Geohash and Google's Open Location Code ("Plus Codes") — that make the same omission Cassini-*style* graticule addressing made, not by historical accident but by deliberate design choice, more than a century after Chapter 2's precedent and squarely inside the era of the polyhedral DGGS family this volume otherwise treats as the modern solution to that same omission.

**Methodological Note**

This chapter introduces five new sources to Section 114.6: Wikipedia (2026a) and Morton (1966) (the latter cited at one remove, via the former, since the 1966 IBM technical report itself was not fetched in this pass) for Geohash; Google Open Location Code contributors (2026), Rinckes (2015), and Wikipedia (2026b) for Open Location Code. No computation is performed; the one quantitative claim below about cell-area variation with latitude follows directly from each scheme's own published encoding rule (interleaved base-2 or base-20 digits of unprojected latitude and longitude), not from a new measurement.

## 5.2 Geohash (2008): Morton order, reinvented and misattributed to no one

Wikipedia (2026a) report that Gustavo Niemeyer devised Geohash in February 2008 while building the geohash.org web service, placing the result directly into the public domain. The encoding itself is a Z-order (Morton) curve: latitude and longitude, each already expressed as an ordinary signed decimal degree, are repeatedly bisected and their bits interleaved, then the interleaved bitstream is regrouped into 5-bit chunks and mapped onto a 32-character alphabet (the digits 0–9 and the lowercase letters b–z, omitting `a`, `i`, `l`, and `o` to avoid both visual ambiguity and accidental word-spelling). Every additional character added to a Geohash string bisects both the latitude and the longitude range of the cell it names, so a Geohash of length $n$ denotes a rectangular cell of angular size $180^\circ/2^{\lceil 5n/2 \rceil} \times 360^\circ/2^{\lfloor 5n/2 \rfloor}$ (up to the fixed bit-order convention), directly in latitude/longitude, with no intervening polyhedral or auxiliary-sphere

step of any kind.

**To Complete / Verify**

**Resolved.** This chapter had cited Morton's report only through Wikipedia (2026a)'s account of it, without independently verifying the report's own existence, title, or abstract. This revision fetches IBM Research's own Technical Paper Search record for it directly (dominoweb.draco.res.ibm.com), confirming the exact title (*A Computer Oriented Geodetic Data Base; and a New Technique in File Sequencing*), sole authorship by G. M. Morton, 1966 publication as an IBM Research Report, and an abstract that itself confirms the Canadian geodetic framing (the report explicitly develops a coordinate system for map or survey data "on any part of the Earth's surface," with "particular reference ... made to Canada"), consistent with Wikipedia (2026a)'s account. Independent citing sources (an OEIS entry, and corrigenda to *Real-Time Rendering*, 2nd edition) separately corroborate the report's exact date as March 1, 1966, and its IBM Ltd., Ottawa institutional origin. The interleaved-bit construction's identity to Geohash's own encoding, and Niemeyer's stated unawareness of Morton's report in 2008, are still sourced to Wikipedia (2026a) alone, since the report's full technical content (as opposed to its title, authorship, and abstract) was not read in this pass.

## 5.3 Open Location Code (2014): the same graticule bisection, renamed

Six years after Geohash, and independently of it, Rinckes (2015) report that a team at Google's Zurich engineering office, crediting Doug Rinckes and Philipp Bunge as the specification's authors (Google Open Location Code contributors, 2026), released Open Location Code (OLC) in late October 2014, with location codes under the system referred to as "Plus Codes." Wikipedia (2026b) record that Google Maps added Plus Code search support the following August (2015). The encoding strategy, per Google Open Location Code contributors (2026), is structurally the same move Geohash makes: latitude (shifted into $[0°, 180°]$) and longitude (shifted into $[0°, 360°]$) are each independently encoded in base 20 using a purpose-built alphabet that avoids vowels (so that no code spells an existing word) and visually similar characters, five digits per coordinate for a default ten-digit code, with a non-significant + inserted after the eighth digit for readability. As with Geohash, every additional digit pair refines both the latitude and longitude extent of the named cell by a further factor of 20; Wikipedia (2026b) give the design rationale explicitly as human-usability (telephone-dictatable, offline-decodable, independent of any street-naming infrastructure) rather than any cartographic property of the resulting cells.

**To Complete / Verify**

Google Open Location Code contributors (2026) names Doug Rinckes and Philipp Bunge as the specification document's authors, while Rinckes (2015), the 2015 announcement, names only Rinckes. This chapter does not resolve the discrepancy; both figures are credited as Google Zurich staff associated with the project, and nothing in this chapter's argument depends on which of the two contributed which part of the design.

## 5.4 The shared property: latitude/longitude bisection, not sphere subdivision

**Key Result**

Geohash and Open Location Code both subdivide the unprojected latitude/longitude rectangle directly, by repeated bisection in each coordinate independently, rather than subdividing a projected image of the sphere. This is exactly the construction Chapter 2 examined as this volume's oldest discrete-addressing precedent, with one difference: that chapter's graticule cells were fixed-size degree-and-minute blocks, while Geohash and OLC cells shrink geometrically with each additional character, giving them the *hierarchical*, arbitrary-precision property Chapter 25 treats as a DGGS design requirement. What neither system gains from this hierarchy, because neither ever leaves the unprojected latitude/longitude rectangle, is any correction for graticule convergence: a Geohash or Plus Code cell's east–west physical extent at fixed angular width shrinks toward the poles by the same $\cos\phi$ factor Chapter 2 already quantified for the bare graticule, while its north–south extent does not, so the cell's physical aspect ratio, and with it its area, is latitude-dependent by construction, in exactly the way every polyhedral DGGS in Part I was built specifically to avoid.

## 5.5 Why both systems accept this, and what it means for this volume's frame

Neither Rinckes (2015) nor Wikipedia (2026a) present this as an oversight. Geohash's own stated purpose, per Wikipedia (2026a), is a sortable, prefix-matchable string key for ordinary database indexes and short shareable URLs; Open Location Code's stated purpose, per Rinckes (2015), is a human-usable substitute for a street address where none exists. Neither goal requires equal-area cells, congruent hierarchy, or any of the properties Chapters 19 and 25 established as OGC DGGS design concerns (Chapter 26): a database index needs only that nearby points share long prefixes, and a spoken or texted address needs only that it be short and hard to mishear, both of which plain latitude/longitude bisection delivers as well as any polyhedral scheme would, at a fraction of the implementation cost. This chapter's point is not that Geohash or OLC are defective DGGS designs — by their own stated goals they are not DGGS designs at all, a category distinction Chapter 26 makes formally — but that they are the clearest evidence available that this volume's central design commitment (project onto a polyhedron or auxiliary sphere *before* tessellating) is a choice made for cartographic reasons specific to equal-area and low-distortion global analysis, not a step every successful discrete geocoding system in production use has found necessary. Two of the most widely deployed discrete location codes in the world — one underlying countless spatial-database indexes, the other built into Google Maps since 2015 — settled the opposite way, and are worse off cartographically, not worse off as products, for it.

**To Complete / Verify**

This chapter does not attempt a quantitative area-variation figure for Geohash or OLC cells at specific latitudes analogous to the Tissot tables computed elsewhere in this volume (e.g. Chapters 79 and 91), since neither system is a projection with a well-defined Jacobian in the relevant sense — both partition unprojected latitude and longitude directly, so the area-

distortion figure is simply $1/\cos\phi$, the bare graticule's own convergence factor already quantified in Chapter 2, and re-deriving it here would duplicate that chapter's computation rather than add to it.

Chapter 6

# Web Mercator: The Efficiency/Accuracy Trade-off in a Projection Everyone Uses

## 6.1 Scope: the trade-off this volume usually poses for DGGS, posed instead for a continuous projection

Chapter 92 and its companion chapters (Chapters 93 to 95) develop this volume's central Part III claim — that DGGS choice trades measurable distortion accuracy against measurable computational efficiency — entirely within the discrete-grid literature: H3 versus S2 versus rHEALPix versus DGGRID, benchmarked cell by cell. This chapter poses the same trade-off for an ordinary continuous projection that almost every reader of this volume has looked at today, deliberately or not: Web Mercator, the projection behind Google Maps, OpenStreetMap, Bing Maps, Mapbox, and effectively every other tile-based web map. Unlike the historical precedents this volume's other "a"-chapters examine (Chapters 2 to 4), Web Mercator's efficiency-over-accuracy trade was not made under wartime urgency a century ago; it was made by a handful of engineers in 2005 and is still, twenty years later, the default rendering surface for a very large fraction of the world's digital cartography. It is, in that sense, the one case in this volume where the cost of distortion is not a historical curiosity but a live design decision made, largely invisibly, on the reader's own behalf every time a map tile loads.

**Methodological Note**

This chapter introduces three new sources to Section 114.6, all fetched and read directly in this pass. The first is Wikipedia contributors (2026), used for the projection's formal definition, its EPSG identifier history (the informal "EPSG:900913" code, the 2008 "Popular Visualisation CRS" EPSG:3785, and the 2009 EPSG:3857 that switched the coordinate reference base from a sphere to the WGS84 ellipsoid while retaining spherical projection formulas), and the documented 2014 NGA advisory notice against the projection's use for authoritative positioning; this source in turn cites Battersby et al. (2014)'s peer-reviewed *Cartographica* analysis of Web Mercator's implications, which this chapter's revised closing todobox reports was subsequently fetched and read directly, upgrading the area-inflation figures below from qualitative to numerically verified. The second and third are two independent web explainers, MapAtlas (2026) and Geography Worlds (2026), used only for the uncontroversial, non-numerically-sensitive claim that Google's 2005 engineering choice was driven by the requirement for square, power-of-two tile pyramids, cross-checked against each other and against Wikipedia contributors (2026)'s independent statement of the same design motivation. No computation is performed in this pass; a future revision could reproduce the area-inflation figures below directly from the projection's own formulas, in the manner of this volume's other numerically verified chapters.

## 6.2 What Web Mercator is: a conformal sphere trick wearing an ellipsoidal datum

Wikipedia contributors (2026) documents that Web Mercator uses the same closed-form spherical Mercator equations that cartographers have used for small-scale maps for centuries, applied uniformly at every zoom level, rather than switching to the more expensive ellipsoidal Mercator formulas that large-scale (city- or province-level) maps conventionally use. The underlying coordinates are nonetheless specified in the WGS84 ellipsoidal datum, the same one this volume already discusses in Chapter 27 and Chapter 28 as the datum underlying most modern DGGS work; Wikipedia contributors (2026) is explicit that projecting ellipsoidal coordinates with spherical formulas makes Web Mercator not quite conformal in the strict sense, though the deviation is too small to be visible by eye. The projection also, per Wikipedia contributors (2026), uses the WGS84 ellipsoid's equatorial radius rather than any compromise mean radius, so the resulting map is very slightly larger than its own stated nominal scale would predict — a small numeric inconsistency that the source attributes to prioritizing implementation simplicity over strict geodetic correctness at every step of the design.

### 6.2.1 The EPSG registry's own initial refusal

The clearest documented statement of professional cartographic objection to Web Mercator that this chapter has located is not a journal article but an institutional refusal: Wikipedia contributors (2026) reports that when Google's and Microsoft's engineers first sought a formal EPSG spatial-reference identifier for the coordinate system underlying their web maps, the EPSG Geodesy subcommittee declined, stating in its own words that it had "reviewed the coordinate reference system used by Microsoft, Google, etc. and believe[d] that it is technically flawed," and that it would "not devalue the EPSG dataset by including such inappropriate geodesy and cartography." The projection circulated for years under an unofficial, self-assigned identifier, "EPSG:900913" — Google spelled with digits — before EPSG relented in 2008 with a spherical-model identifier, EPSG:3785, explicitly flagged as "not an official geodetic system," and then, in 2009, with the ellipsoidal-datum identifier still in use today, EPSG:3857 (Wikipedia contributors, 2026). The same source documents a second, later, institutional rejection along the same lines: in 2014 the U.S. National Geospatial-Intelligence Agency issued a formal advisory notice declaring Web Mercator "unacceptable for any official use" by the Department of Defense, on the grounds that confusing it with the standard (ellipsoidal) Mercator during coordinate conversion can produce ground-position errors of up to 40 km (Wikipedia contributors, 2026). This is the same institutional lineage — NGA, the successor to the Army Map Service — that this volume's Chapter 4 credits with building MGRS on top of UTM precisely because ordinary users could not be trusted to apply geodetic corrections correctly by hand; six and a half decades later, the same agency is declining to endorse a different projection for the analogous reason.

## 6.3 Why it was adopted anyway: square tiles, not scientific accuracy

Against that documented professional objection, MapAtlas (2026) and Geography Worlds (2026) agree, independently of each other and of Wikipedia contributors (2026), on the specific engineering reason Google's mapping team adopted the projection in 2005: a conformal cylindrical projection with equally spaced vertical meridian lines produces, at every zoom level, a perfectly square world map whose side length doubles from one zoom level to the next, let-

ting the whole map be sliced into a power-of-two pyramid of square image tiles (zoom 0 one tile, zoom 1 four tiles, zoom 2 sixteen tiles, and so on) with no leftover fractional tiles at any edge (MapAtlas, 2026). Because the spherical Mercator formula requires no per-request scale-factor computation, the same sources report that using it at every zoom level, rather than the costlier ellipsoidal formula that large-scale maps conventionally use, was chosen specifically to minimize server- and client-side computation for a system that had to render and re-render millions of tiles on demand (MapAtlas, 2026; Geography Worlds, 2026). Wikipedia contributors (2026) independently confirms the same design rationale from the geodetic side: the projection was built to be computationally cheap first, and only secondarily to be an accurate representation of the WGS84 ellipsoid it is nominally defined against.

**Key Result**

Web Mercator is, in the vocabulary this volume's Part III applies throughout to DGGS, a documented case of the identical trade this volume's discrete-grid chapters spend hundreds of pages quantifying cell by cell: a design that sacrifices a formally provable accuracy property — here, strict conformality and any claim to equal-area-ness, rather than any DGGS's ring-placement law or spherical-IPQ ceiling (Chapter 100) — for a large, measurable gain in computational and architectural simplicity, namely the existence of a clean power-of-two square-tile pyramid. The difference from this volume's DGGS material is institutional rather than mathematical: no explicit distortion-versus-efficiency accounting of the kind this volume's own DGGS chapters compute was carried out before the trade was made, and the two professional bodies that did formally evaluate the trade after the fact — EPSG's Geodesy subcommittee and the NGA — both declined to endorse it, yet adoption by two of the world's largest mapping platforms settled the question in practice regardless. Where this volume's DGGS chapters treat the efficiency/accuracy trade-off as a design parameter to be optimized and reported, Web Mercator's twenty-year dominance is evidence that, at sufficient user scale, the trade can also simply be made once, informally, by an engineering team, and then never revisited by the platforms that inherit it.

## 6.4 The area-inflation cost, quantified qualitatively

Because Web Mercator is a (near-)conformal cylindrical projection, it shares the ordinary Mercator's signature failure mode that this volume's Chapter 30 already formalizes via Tissot's indicatrix: area inflation that grows without bound toward the poles, with the projection undefined at the poles themselves. Wikipedia contributors (2026) reports the specific practical consequence that every major Web Mercator implementation adopts the same fixed cutoff latitude, $\pm 85.0511°$, chosen because it is exactly the latitude at which the (square, power-of-two) projected world map itself becomes a perfect square in projected coordinates; areas closer to the poles than this are simply not rendered by services such as Google Maps, rather than being shown at their true (very large) projected extent. This chapter does not itself recompute the area-inflation factor at that cutoff latitude or elsewhere — doing so would only reproduce, for one specific conformal cylindrical projection, the same closed-form Tissot analysis Chapter 30 already carries out in general; the point this chapter adds is institutional rather than mathematical, namely that this particular, well-quantified area distortion is the one every reader of a modern web map silently accepts, in exchange for a tiling scheme, every time the map loads.

**To Complete / Verify**

**Resolved.** This chapter had cited Battersby et al. (2014) only at one remove, via Wikipedia contributors (2026)'s citation of it, without fetching the original. This revision fetches the article directly and can now upgrade the area-inflation claim from qualitative to numerically verified. Battersby et al. (2014)'s Table 4 gives the computed area of Alaska under eight projections: equal-area schemes agree closely with the true figure (Albers Equal Area Conic and Alaska Albers both give $1{,}517{,}411.29$ km$^2$, Sinusoidal $1{,}517{,}422.58$ km$^2$, Lambert Azimuthal Equal Area $1{,}517{,}412.27$ km$^2$, against the US Census's real figure of $1{,}477{,}953$ km$^2$ – the residual few-tenths-of-a-percent gap across all four equal-area schemes is itself informative, and is not resolved in Battersby et al.'s own text), while Mercator gives $8{,}181{,}068.03$ km$^2$ and Web Mercator $8{,}191{,}239.76$ km$^2$ – both roughly $5.5\times$ the true area, with Web Mercator inflating area a further $0.12\%$ beyond Mercator itself at this specific high latitude. The article also supplies a directly comparable km/degree-of-longitude distortion table (its Table 3) reaching a 77 km discrepancy at Alaska's own latitudes, corroborating rather than merely asserting the qualitative claim this section made before this revision.

Chapter 7

# Compromise Projections Before Peters

## 7.1 Scope: a cartographic tradition, not a single controversy

Chapter 35 traces this volume's own distortion-cost genealogy along a mathematical line — Tissot, Chebyshev, Milnor, Peters, Goldberg-Gott — in which the 1973 Peters controversy figures as a specifically *political* reframing of the equal-area-versus-conformal choice. This chapter looks upstream of that controversy, at a parallel, largely apolitical, institutional tradition: the sequence of "compromise" world-map projections — projections that are deliberately neither conformal nor equal-area, adopted by name for exactly that reason — that professional cartographers and one major map-publishing institution, the National Geographic Society (NGS), worked through across most of the twentieth century, decades before Peters's 1973 intervention and independently of it. Where Chapter 2 found a nineteen-century-old precedent for discrete addressing and Chapter 3 found a century-old precedent for regional distortion control, this chapter's precedent is narrower and closer to the volume's own title: an explicit, self-described practice of trading away both of the two properties this volume's later chapters spend the most effort trying to preserve or restore.

**Methodological Note**

This chapter draws on one source already in Section 114.6 but newly fetched in full for this pass — Snyder (1987)'s §29, "Van der Grinten Projection" (pp. 239–242 of the 1987 printing), read via a complete section-by-section HTML mirror of the Working Manual — and one new, explicitly tertiary source, Wikipedia contributors (2026), used only for the Robinson and Winkel Tripel projections, for which no primary source was located or fetched in this pass, and cross-checked against Esri's ArcGIS Pro projection documentation. No computation is performed.

## 7.2 Van der Grinten (1898/1904): the first widely adopted compromise

Snyder (1987) reports that Alphons J. van der Grinten, of Chicago, presented four world projections in a 1904 German geographical-journal article, all patented in the United States that same year; the best known of the four, showing the whole Earth enclosed in a circle with a straight Equator and central meridian and all other meridians and parallels as circular arcs, had actually been invented earlier, in 1898. Snyder (1987) is explicit, as this volume's own house style requires of every projection it discusses, that the Van der Grinten projection is neither conformal nor equal-area by design: van der Grinten himself, per Snyder (1987), described it as combining the familiar visual appearance of the Mercator with the curved meridians of the equal-area Mollweide projection, without claiming either projection's defining property for the result.

**To Complete / Verify**

**Narrowed, still not resolved.** Snyder (1987) states that the National Geographic Society used the Van der Grinten projection for its standard world map *since 1943*, against several tertiary sources giving 1922. This revision fetched Snyder (1987)'s own Van der Grinten section directly (rather than citing it at one remove) and confirms the 1943 wording verbatim, plus surrounding detail not previously reported here: the projection was patented by van der Grinten in the United States in 1904, was separately adopted by the U.S. Department of Agriculture as a base map for economic data in the 1940s, and was subsequently reused by USGS for a four-sheet 1970 subsea-minerals map series. This strengthens Snyder (1987) as a directly-read primary cartographic-history source rather than weakening the 1922 figure: the 1922 date is not a tertiary artifact but is stated by several independent sources, including, as of this revision, National Geographic Education (2026) — the National Geographic Society's own current education website, which states 1922 explicitly. The discrepancy therefore now stands between two authoritative-seeming sources (Snyder's USGS professional paper versus NGS's own present-day account of its history), not between a primary source and a tertiary encyclopedia; this chapter discloses rather than adjudicates it, and a future revision should locate a mid-20th-century NGS map or press account stating the adoption date contemporaneously, which would settle the matter more directly than either retrospective source can.

## 7.3 Robinson (1963): compromise by eye rather than by equation

Per Wikipedia contributors (2026), cross-checked against Esri's ArcGIS Pro documentation, Arthur H. Robinson designed his eponymous projection in 1963 at the request of the Rand McNally map-publishing company, and did so by graphic design — iteratively sketching and adjusting a graticule until it looked right, then tabulating the resulting coordinates for later interpolation — rather than by first choosing a distortion property and deriving equations from it, the order every other projection this volume discusses in detail follows.

**Key Result**

The National Geographic Society's own successive choices of standard world-map projection, read across Snyder (1987) and Wikipedia contributors (2026) together, form a single institution's repeated re-litigation of the identical equal-area-versus-conformal trade-off across three-quarters of a century, entirely outside the Peters controversy's own terms: Van der Grinten from (per Snyder, 1987) 1943 to 1988, Robinson from 1988 to 1998, and Winkel Tripel from 1998 to the present. Each change was justified, per Wikipedia contributors (2026), on cartographic rather than political grounds — the 1998 move to Winkel Tripel is described there as motivated by reduced polar-region distortion of landmasses, not by any equal-area objection of the kind Peters raised against conformal world maps in 1973 (Chapter 35). This is the same underlying tension this volume's own DGGS efficiency-versus-accuracy chapters (Part III) formalize quantitatively, worked out here instead by three successive rounds of institutional trial and replacement rather than by a single closed-form optimization.

## 7.4 Winkel Tripel (1921/1998): compromise by explicit averaging

Per Wikipedia contributors (2026), Oswald Winkel proposed his Tripel projection in 1921, decades before either its NGS adoption or the Peters controversy it now postdates by half a century in the opposite direction. Unlike Van der Grinten's geometric construction or Robinson's graphical trial-and-error, the Winkel Tripel is constructed, per Wikipedia contributors (2026), by taking the arithmetic mean of the projected coordinates given by two other, individually named projections — the Aitoff projection and the equirectangular (plate carrée) projection — for every point on the graticule. The name itself, per Wikipedia contributors (2026), is German for "triple," referring not to three source projections but to Winkel's stated goal of moderating three kinds of distortion (area, direction, and distance) at once rather than eliminating any one of them, the most explicit institutional statement, of the three projections this chapter treats, that a compromise projection's purpose is to spread a fixed distortion budget across several properties rather than to zero out any single one.

**To Complete / Verify**

**Citation strengthened, full text still unread.** No primary source for Winkel's 1921 article (*Neue Gradnetzkombinationen*, in *Petermanns Mitteilungen*) was successfully fetched in the original pass; the English-translation mirror once hosted at `csiss.org` is now confirmed dead by an independent map-projection history blog (kartenprojektionen.de, May 2022 update note), not merely inaccessible to this volume's own fetch attempt. That same blog and a 2024 peer-reviewed article (Lapaine, 2024, *KN–Journal of Cartography and Geographic Information*) both independently cite the identical full reference – *Petermanns Mitteilungen*, vol. 67, December 1921, pp. 248–252 – corroborating the tertiary Wikipedia citation exactly rather than merely repeating it, and the cartography-history blog adds one further primary-adjacent detail: Winkel's own original drawing is inscribed "Erstmalig entworfen am 15. September 1920" (first designed 15 September 1920), a year before publication. The article's own text remains unread in this pass; the numeric claim above (arithmetic mean of Aitoff and equirectangular) rests on Wikipedia contributors (2026) alone, still unupgraded to a primary-sourced figure.

## 7.5 Closing: compromise as a third strategy, not a fallback

Read against Chapter 35's Tissot-to-Goldberg-Gott line, this chapter's three projections are not failed attempts at conformality or equal-area-ness that Peters's controversy later exposed as inadequate; per every source fetched for this chapter, each was adopted, and in Robinson's and Winkel's cases explicitly designed, *as* a compromise, with the trade-off stated as the point rather than as a limitation to be apologized for. Van der Grinten's 1898 construction, Robinson's 1963 graphical method, and Winkel's 1921 averaging rule give this volume's own cost-of-distortion framing a third historical strategy, alongside the graticule's discrete addressing (Chapter 2) and the transverse Mercator's regional zoning (Chapter 3): rather than tessellate the sphere finely, or restrict a single projection to a small region, hold the whole globe in one projection and let no single distortion property win outright. All three strategies precede this volume's own DGGS material by decades to millennia; none of them, on the evidence gathered in this chapter and its two companions, was designed with knowledge of, or in reaction to, the other two.

Chapter 8

# Interrupted and Composite Projections: Discontinuity as a Fifth Strategy

## 8.1 Scope: splitting the globe on purpose

Chapter 7 catalogs Van der Grinten, Robinson, and Winkel Tripel as a twentieth-century tradition of *whole-globe* compromise projections, each mapping the entire sphere onto one uninterrupted sheet while deliberately sacrificing both conformality and equal-area-ness. This chapter examines a different, and in one precise sense more radical, twentieth-century response to the same underlying problem: rather than accept distortion everywhere in exchange for global continuity, deliberately *cut* the sphere along extra seams — beyond the one east-west seam every world map requires — so that each resulting piece is small enough to carry comparatively little distortion, at the cost of showing the globe as several disconnected lobes rather than one continuous sheet. The archetypal instance, and this chapter's main case study, is John Paul Goode's 1923 homolosine projection; but the same interruption principle connects backward, within this volume's own material, to the polyhedral tradition Chapter 22 already covers (Cahill's octahedral Butterfly Map and Fuller's Dymaxion, both already discussed there as interrupted-by-construction polyhedral unfoldings) and forward to composite-lettering schemes this volume has not otherwise treated.

**Methodological Note**

This chapter introduces three new sources to Section 114.6, all fetched and read directly in this pass. The first two are Wikipedia contributors (2026) and its companion article Wikipedia contributors (2026), cross-checked against each other and, for the projection's technical definition, against Esri (2026), a tertiary technical-documentation source of the same kind this volume already uses elsewhere (Chapter 20 for PROJ contributors, 2026). No primary 1923–1925 Goode source (his own *Annals of the Association of American Geographers* article, cited by Wikipedia contributors, 2026 but paywalled at the location checked in this pass) was fetched directly; every claim in this chapter attributed to Goode's own reasoning is cited via the secondary and tertiary sources above, not independently verified against his original text. No computation is performed.

## 8.2 From Mollweide interrupted to homolosine: Goode's 1916–1923 sequence

Wikipedia contributors (2026) and Wikipedia contributors (2026) agree on the same two-stage development: in 1916, University of Chicago geographer John Paul Goode experimented with cutting the existing, already equal-area, Mollweide projection into separate lobes along several meridians rather than just one, splitting the globe's east-west continuity deliberately in order to let each land-bearing lobe sit closer to its own locally best central meridian and so shrink the

shape distortion any single lobe had to carry. Satisfied with the resulting interruption scheme, Goode then, in 1923, built a new composite projection specifically to carry that same interruption pattern: the homolosine, formed by grafting the sinusoidal projection's equatorial band (between $\pm 40°44'11.8''$ latitude) onto Mollweide's polar caps at the two parallels where the two source projections' scales along the parallel happen to match (Wikipedia contributors, 2026; Esri, 2026). Wikipedia contributors (2026) reports that the name itself is a portmanteau, fusing "homolographic" (an older name for the equal-area Mollweide) with "sinusoidal," and that because both source projections are individually equal-area, the graft preserves that property exactly across the whole composite — at the documented cost of a visible "kink" in the meridians at every latitude where the two source projections are stitched together, since nothing in the construction requires their derivatives to match at the seam, only their scale.

**Key Result**

Read against this volume's own vocabulary, Goode's homolosine is a composite projection in exactly the same structural sense this volume already applies to composite DGGS (Chapter 20's cube-face constructions, or the dual-scheme material in Chapter 59): two separately well-understood, individually equal-area mappings, joined along a chosen seam chosen specifically because the two pieces' scales agree there, producing a single projection whose global equal-area property is inherited piecewise rather than derived from one closed-form construction. The elective, beyond-mathematical-necessity interruption that Goode then applies on top of this composite — cutting the oceans into separate lobes so that continental landmasses stay comparatively whole — is a second, independent design choice from the homolosine construction itself: Wikipedia contributors (2026) is explicit that "these extra interruptions do not create a new projection," they merely rearrange an existing one, in the same sense that choosing which DGGS cells to render or query does not change the DGGS's own underlying tessellation.

## 8.3 Interruption as a general strategy: from globe gores to myriahedral maps

Wikipedia contributors (2026) situates Goode's oceanic interruption scheme within a much older and, at its far end, much more elaborate family of deliberately split world maps. At the older end, the source documents that globe gores — printed map segments designed from antiquity onward to be pasted onto the curved surface of a physical globe — are themselves a regular interruption scheme, either along the equator or, in polar "rosette" form, around a pole; this is the same underlying idea of tiling a sphere with flat, individually low-distortion pieces that Chapter 22 already traces through Cahill's and Fuller's polyhedral world maps, which Wikipedia contributors (2026) explicitly identifies as interrupted maps in this same sense, generalizing the gore principle from a sphere-hugging strip to a full polyhedral face. At the newer, more elaborate end, Wikipedia contributors (2026) documents two twentieth- and twenty-first-century extensions this volume has not previously discussed: Athelstan Spilhaus's interruption scheme, which cuts the globe along continental boundaries instead of oceanic ones so as to show the world's oceans as a single connected body of water, the exact inverse emphasis from Goode's land-preserving scheme (Spilhaus, 1983); and Jarke J. van Wijk's "myriahedral" projections (van Wijk, 2008), which push the same principle to its computational extreme by interrupting the globe along many thousands of short polygon edges simultaneously. In van Wijk's own construction, the globe is projected onto a fine polyhedral mesh, each mesh edge is weighted and labeled as a fold or a cut by computing a maximal-weight spanning tree over the mesh's dual graph (using Prim's algorithm), and the mesh is then unfolded along the fold

edges alone; three weighting schemes are offered, based respectively on the plain graticule, on recursive subdivision of a Platonic solid, and on meshes aligned to continental boundaries so that cuts fall preferentially through oceans rather than through land. Each individual polygon face is small enough that its own area and angular distortion are close to negligible, so the resulting maps are, in the author's own words, "(almost) conformal and conserve areas," at the cost of a very large and visually fragmenting number of interrupts rather than a small, hand-chosen set as in Goode's homolosine.

**To Complete / Verify**

**Substantially resolved.** This chapter's account of the Spilhaus and myriahedral extensions had rested entirely on Wikipedia contributors (2026)'s brief mention of each, without fetching either primary source, and had also misnamed the myriahedral projections' inventor as "Jack van Wijk"; this revision fetches van Wijk (2008)'s full peer-reviewed text directly (the author's own hosted PDF) and corrects the name to *Jarke J. van Wijk*, as the paper itself signs it. The technical description above (spanning-tree fold/cut labeling, the three weighting schemes, and the "almost conformal and area-preserving" characterization) is now sourced to the paper's own Method and Discussion sections rather than to a tertiary summary, replacing the qualitative-only account this chapter previously offered; van Wijk's own paper further reports, consistent with Chapter 22's account of Fuller's Dymaxion map, that an unconstrained optimal-orientation search over Platonic solids reproduces Fuller's continent layout for every solid except the tetrahedron, and separately cites Spilhaus (1983) by full reference (*Proceedings of the American Philosophical Society* 127(1), 50–60, 1983) for the Spilhaus ocean-centered map this chapter cites, resolving that citation as well. Spilhaus's original article's own text remains unread in this pass, sourced only via van Wijk's citation of it and Wikipedia's summary; no quantitative distortion figure comparing either scheme numerically to Goode's homolosine was located, since van Wijk's paper characterizes distortion qualitatively (as "almost" zero within each face) rather than with a summary statistic comparable to Chapter 30's figures.

## 8.4 Closing: discontinuity as this volume's fifth historical strategy

This chapter completes a set of five distinct historical strategies this volume's "a"-series chapters have now catalogued for handling the cost of distortion, each predating this volume's own DGGS material by decades to centuries and, on the evidence gathered across all five chapters, developed independently of one another: discrete graticule addressing (Chapter 2), regional projection zoning and its alphanumeric indexing sequel (Chapters 3 and 4), whole-globe compromise (Chapter 7), single-property functional restriction (Chapter 10), and now deliberate discontinuity. Where the first four strategies each hold some property fixed — a discrete address, a narrow zone, a whole-globe compromise budget, or one exactly-preserved functional property — while letting the map's own continuity or shape absorb the remaining cost, interruption instead holds distortion-per-lobe low by giving up the sphere's own topological continuity outright, accepting a map that is honest about being several separate pieces rather than one. Goode's own stated motivation, per Wikipedia contributors (2026), was exactly this trade, offered as an equal-area alternative to the Mercator projection for "portraying global areal relationships"; that his solution required literally tearing the map apart to get there is, read against this volume's title, as direct a statement as any of this volume's other historical precedents that the cost of distortion has always had to be paid in *some* currency, continuity included.

Chapter 9

# A Conformal Square, a Century Early: Peirce and Guyou

## 9.1 Scope: seamless tiling as a nineteenth-century conformal result

Chapter 8 examined how twentieth-century cartographers cut the sphere's continuous surface into separately mapped, edge-matched pieces to control distortion regionally; this volume's DGGS chapters (Chapters 19 and 21) examined how the same seamless-tiling goal is met today by projecting each face of a regular polyhedron separately, with the polyhedron's own combinatorics guaranteeing exact edge matching. This chapter's subject predates both of those solutions by decades: a single, exactly conformal projection of the whole sphere onto a square that tiles the plane seamlessly in every direction with no polyhedron, no interruption, and no edge-matching construction at all beyond the projection formula itself.

**Methodological Note**

This chapter introduces six new sources to Section 114.6: Peirce (1879) and Guyou (1887) (both cited only via the tertiary/documentation sources below, since neither nineteenth-century original was fetched directly in this pass); Wikipedia (2026c), Wikipedia (2026d), and Esri (2026b), of the same tertiary/documentation kind already used elsewhere in this volume for historical projections whose primary sources were not independently fetched (Wikipedia contributors, 2026 in Chapter 7). No computation is performed; the one quantitative comparison below (area-exaggeration footprint against Mercator and stereographic) is transcribed from Wikipedia (2026d), not re-derived.

## 9.2 Peirce (1879): a square built by Schwarz–Christoffel mapping

Per Wikipedia (2026c) and Esri (2026b), Charles Sanders Peirce — working at the United States Coast and Geodetic Survey, and better known today as a founder of American pragmatist philosophy than as a cartographer — published in 1879 a conformal projection of the sphere onto an unfolded square dihedron. Each of the sphere's eight octants maps to one isosceles right triangle; in the projection's normal aspect, four such triangles tile into a central square holding the entire Northern Hemisphere, while the four Southern Hemisphere triangles fold out to the corners, producing the quincunx arrangement — a central square flanked by four quarter-squares, echoing the five-point pattern on a die's five face — that gives the projection its name. Wikipedia (2026c) report that Peirce built the construction on H. A. Schwarz's 1869 conformal mapping of a disc onto an $n$-sided polygon (the Schwarz–Christoffel mapping, independently also due to Christoffel in 1867): a stereographic projection of the sphere onto a plane, followed by a Schwarz–Christoffel map of that plane's disc onto a square.

**Key Result**

The resulting square is, per Esri (2026b), conformal everywhere except at four isolated points along the equator — the midpoints of the square's four edges — where the mapping is singular and angles are not preserved. Away from those four points, the Peirce quincuncial is exactly angle-preserving, and, per Wikipedia (2026c), tiles the infinite plane seamlessly in every direction: adjacent copies of the square match along every edge with no discontinuity, a global seamless-tiling property this volume's DGGS chapters achieve only by explicit polyhedral construction (Chapter 21) or by deliberate interruption design (Chapter 8), achieved here instead as a direct analytic consequence of one conformal mapping formula, with no polyhedron and no cut-and-reassemble step at all.

## 9.3 Guyou (1887) and Adams (1925): the same formula, reoriented

Per Wikipedia (2026d), the French naval officer and scientist Émile Guyou proposed in 1887 an oblique aspect of Peirce's construction — obtained, per that source, by rotating the coordinate axes $45°$ before the stereographic step and rotating the result back afterward — mapping each hemisphere onto its own square, so that the whole sphere becomes a $2{:}1$ rectangle rather than Peirce's single square holding one hemisphere at the centre. Wikipedia (2026d) report that a transverse aspect of the same underlying mapping was proposed later still, by Oscar S. Adams in 1925. All three projections — Peirce's polar aspect, Guyou's equatorial aspect, and Adams's transverse aspect — are, per Wikipedia (2026c), the identical Schwarz–Christoffel construction under three different choices of which point on the sphere is sent to the mapping's centre, in exactly the sense Chapter 21 established for polyhedral DGGS orientation choices: one underlying mathematical object, three cartographically distinct results, differing only in where the singular points and the seams fall relative to the geography being mapped.

**To Complete / Verify**

**Resolved.** Wikipedia (2026c) reports 1879 as Peirce's publication year, consistent with Esri (2026b)'s independent documentation-page account, but had noted that a single secondary source (Lee, 1976) instead gives 1877. This bibliographic revision fetched Peirce (1879)'s own indexed publication record directly (JSTOR stable/2369491) rather than a third tertiary opinion: it confirms *American Journal of Mathematics*, Vol. 2, No. 4, issue dated December 1879, pp. 394–396, matching both previously consulted sources exactly and independently of them. The 1877 figure in Lee (1976) is therefore treated as the erroneous outlier, on the strength of the journal's own indexed issue date rather than merely a 2-against-1 count of secondary sources. This still does not explain *why* Lee (1976) reports 1877 (a transcription error, a confusion with Peirce's earlier related work, or something else is not established here), but the publication-year question itself is now closed.

## 9.4 A quantified distortion budget, a century before this volume's own

Wikipedia (2026d) report one further figure worth setting directly beside this volume's own Tissot-based distortion budgets (Chapters 31, 79 and 91): the region of the Guyou projection where scale exaggeration reaches double its central value covers only about $9\%$ of the sphere's area, against roughly $13\%$ for the Mercator projection and $50\%$ for the stereographic projection,

over the same hemisphere. This is precisely the kind of concrete, source-reported distortion accounting Chapter 92 argues every scheme in this volume should be judged by, produced here nineteen years before the Mercator figure it is compared against had even been formally superseded as the default world map (Chapter 7), and a full century before the DGGS family this volume otherwise treats as the modern approach to the same accounting.

## 9.5 Closing: a precedent this volume's polyhedral chapters do not supersede

The Peirce/Guyou/Adams family is not a competitor to the ISEA-family DGGS this volume examines in detail: none of the three offers a hierarchical, addressable cell structure, and Esri (2026b) notes that area distortion, while excluded from the angle-preserving property, still grows steadily from the square's centre outward, so no member of the family is equal-area either. What the family demonstrates, standing beside Chapter 7's Van der Grinten/Robinson/Winkel Tripel lineage and Chapter 2's graticule precedent, is a third historical route to one of this volume's recurring goals — a seamlessly tileable, globally consistent map — reached by a single closed-form conformal mapping rather than by a polyhedron's combinatorics or an interruption scheme's cut lines, and reached decades before either of those two later solutions existed in any form.

Chapter 10

# Straight Lines That Are Actually Shortest: Azimuthal Projections for Navigation and Aviation

## 10.1 Scope: a functional, not a mathematical, history

The azimuthal equidistant projection already appears twice elsewhere in this volume in a purely mathematical role: Chapter 34 treats it as the unique optimum of a Chebyshev-type minimax problem among conformal maps, and Chapter 35 traces its role in the Goldberg–Gott/Gott-Wagener flexion-score genealogy, where a double-sided disk map built on this same projection reaches that survey's lowest reported score (0.881). This chapter, and the closely related gnomonic projection it also treats, are instead examined here for a functional property that has nothing to do with either conformality or equal-area-ness: the gnomonic projection is the unique projection onto a plane under which every great circle on the sphere — every shortest path between two points — appears as a straight line, and the azimuthal equidistant projection is the unique projection under which every straight line drawn *from its center point* has length and direction exactly proportional to the true great-circle distance and initial bearing from that center. Both properties were put to sustained practical use centuries before either projection entered this volume's other, purely distortion-theoretic, discussions: gnomonic charts for ocean and, later, transpolar air navigation, and the azimuthal equidistant for a symbolic case this chapter also covers, the United Nations flag.

**Methodological Note**

This chapter introduces four new sources to Section 114.6. The first is Cotter (1976), a peer-reviewed *Journal of Navigation* article fetched and read in full, covering the sixteenth- through nineteenth-century history of great-circle sailing and its gnomonic-chart technique in detail, including primary quotations it reproduces from John Davis's 1595 *Seamans Secrets* and from Atkinson's eighteenth-century *Epitome of the Art of Navigation*. The second is a search-result summary (the full page was blocked to direct fetching) of a Library of Congress item description (Library of Congress, 2026) for Charles Lindbergh's own annotated 1927 gnomonic transatlantic-flight chart, used only for the non-numerically-sensitive fact of the chart's use and for one short quotation from Lindbergh's own 1953 memoir that the Library of Congress's description itself reproduces. The third and fourth are Wikipedia contributors (2026) and Geography Realm (2023), two independently written sources on the United Nations flag and emblem, cross-checked against each other for the emblem's 1946 adoption date and its attribution to designer Donal McLaughlin. No computation is performed; this chapter, like this volume's other historical "a" chapters, is a sourcing exercise rather than a numerical one.

## 10.2 The gnomonic projection: straight lines as shortest paths, from Renaissance globes to Lindbergh's chart

Cotter (1976) traces documented interest in great-circle (as opposed to easier rhumb-line) sailing back to Pedro Nunez's 1537 *Tratado em defensam de carta de marear*, which states explicitly that a ship's most direct course is always along a great-circle arc, and to a chain of sixteenth- and seventeenth-century English navigational writers — William Bourne, John Davis, Richard Norwood — who advocated navigating by a physical globe and a brass "quadrant of altitude" precisely because no flat chart of the period could represent a great circle as anything but a curve. The source quotes Davis's 1595 *Seamans Secrets* at length identifying "great circle navigation" as "the chiefest of all the 3 kinds of sayling," and documents that the practical method of the period was to repeatedly re-orient a globe, read off a new great-circle bearing to the destination every hundred miles or so, and sail a short rhumb-line leg on that bearing — in effect approximating the great circle by a polyline of short loxodromic segments (Cotter, 1976). Cotter (1976) reports that this practice fell out of use once Mercator sailing with logarithmic tables became standard, and that a genuine *gnomonic-chart* method — plotting the great-circle route as a single straight line on a gnomonic projection, then transferring a series of points from that line onto a Mercator chart to obtain the curved rhumb-line-composite track sailors actually steered — is documented in detail in Atkinson's eighteenth-century *Epitome of the Art of Navigation*, which the source quotes giving a full worked example (Lizard Point to Barbados) of the construction, including the explicit instruction that positions found gnomonically may then be "prick'd down" onto the working Mercator chart (Cotter, 1976).

Cotter (1976) dates a nineteenth-century revival of active great-circle-route planning to the 1840s, driven by Lieutenant Matthew Fontaine Maury's Atlantic wind-and-current charts and by the arrival of steamships no longer constrained to sail only on wind-favorable headings, and credits Lieutenant Henry Raper's 1840 *Practice of Navigation and Nautical Astronomy* with giving the gnomonic-chart method its standard nineteenth-century form. The same underlying property — great circles as straight lines — carried the technique directly into twentieth-century aviation without any change of projection: the Library of Congress's own description of Charles Lindbergh's personal 1927 transatlantic flight chart (Library of Congress, 2026) identifies it as a gnomonic-projection chart that Lindbergh called the "nugget of gold" he found in a San Pedro shop while preparing for the flight, and quotes his 1953 memoir *The Spirit of St. Louis* describing the moment directly: "I drew a straight line between New York and Paris on the gnomonic projection" (Library of Congress, 2026).

**Key Result**

The gnomonic projection's four-century documented run — from sixteenth-century globe-and-quadrant sailing, through Atkinson's eighteenth-century worked chart method, to Lindbergh's own 1927 straight-edge construction — is a case this volume has not yet examined of a projection whose entire historical adoption rests on a single geometric property (great circles map to straight lines) completely orthogonal to the equal-area/conformal axis this volume's other chapters track. The gnomonic projection is neither conformal nor equal-area, and its area and angular distortion both grow without bound away from the tangent point, exactly the kind of distortion profile that would disqualify it from any of this volume's own distortion-minimizing DGGS comparisons; navigators and, later, pilots used it anyway, for four hundred years, because the one property it does preserve exactly — the straightness of shortest paths — was worth more operationally than any global distortion bound.

## 10.3 Azimuthal equidistant: distance-true from one point, and the United Nations flag

Where the gnomonic projection is useful for finding a great-circle route, the azimuthal equidistant projection is useful for reading off distances and bearings from one fixed reference point once a route is known, and Wikipedia contributors (2026) documents its best-known non-navigational use: the emblem at the center of the flag of the United Nations, adopted by the General Assembly on 7 December 1946 (the flag itself following on 20 October 1947), depicts an azimuthal equidistant projection of the world centered on the North Pole, oriented to the International Date Line, extending to 60° south latitude and marked with five concentric distance circles, enclosed in a wreath of two stylized olive branches. Geography Realm (2023) independently confirms the same design details and attributes the emblem to Donal McLaughlin, a designer who had worked in wartime graphics for the Office of Strategic Services (the CIA's wartime predecessor); Wikipedia contributors (2026) adds, without this chapter being able to verify the claim against any further primary source in this pass, that McLaughlin's specific choice of the azimuthal equidistant projection was influenced by the wartime cartography of Richard Edes Harrison, whose *Fortune* and *Life* magazine maps had popularized polar-centered azimuthal views of a strategically interconnected world during the Second World War. Geography Realm (2023) reports that an earlier, April 1945 version of the same emblem had instead centered the projection so as to place the United States near the map's center, and that the design was deliberately altered before the December 1946 adoption so that no member state would appear privileged by the projection's own center point — a use of azimuthal equidistant's defining distance-true property (accurate only from the center outward) turned into a diplomatic argument for choosing a politically neutral center, the North Pole, rather than for choosing the projection itself.

**To Complete / Verify**

**Partially resolved.** The two-version design history (an April 1945 US-centered emblem, replaced before the December 1946 adoption by a politically neutral North-Pole-centered one) had rested on Geography Realm (2023) alone. This revision locates two independent corroborations. First, ICAO (2026), an ICAO postal-history document, reproduces, in quotation marks, wording it attributes directly to the UN Secretary-General's own *Report of the Secretary-General, 15 October 1946*: "a map of the world on a north polar azimuthal equidistant projection inscribed in a wreath of crossed conventional branches of an olive tree," extending to 60° south latitude with five concentric circles – matching this chapter's description exactly and citing General Assembly Resolution 92(I) (7 December 1946) for formal adoption, both details this chapter had sourced only from Wikipedia and Geography Realm. Second, Bourneuf (2024), an independent UN-affiliated account citing UN Archives Geneva material, corroborates the first-version/second-version design change and adds a detail this chapter had not reported: the earlier version's southern cutoff, often attributed in popular accounts to a deliberate exclusion of Argentina, was instead explained by McLaughlin himself as an artifact of the Rand McNally base map he worked from simply not extending that far south – a correction to a common claim that this chapter does not itself repeat, but is worth flagging since Geography Realm (2023) and other tertiary accounts of this same emblem sometimes do repeat it. The Richard Edes Harrison influence claim, by contrast, remains uncorroborated: it still rests on Wikipedia contributors (2026) alone, and no biography of McLaughlin or primary OSS-graphics-division source was located for it in this pass.

## 10.4 Closing: a fourth strategy, orthogonal to the volume's usual axis

Alongside the three historical strategies Chapter 7 already catalogs for handling the cost of distortion — discrete addressing (Chapter 2), regional zoning (Chapters 3 and 4), and whole-globe compromise (Chapter 7) — this chapter's gnomonic and azimuthal equidistant material adds a fourth, orthogonal one: choose the one distortion property a specific functional task actually needs (straight great circles for route planning, or true distance-and-bearing from one fixed point), accept unbounded distortion in every other respect, and restrict the projection's use to exactly the task that single property serves. Every source this chapter cites documents users doing precisely that — sailors and Lindbergh alike used gnomonic charts only to *find* a route, then re-plotted it on a Mercator chart to actually navigate by it, and the United Nations uses its azimuthal equidistant emblem only as a symbol, never as an operational world map. None of this volume's later Part I or Part III material evaluates either projection by any of this volume's own accuracy metrics, since neither was ever intended to minimize distortion in the aggregate sense those metrics measure; the two projections earn their long institutional lives by solving one narrow problem exactly, not by trading off many problems at once.

Chapter 11

# Reproducing the Five-DGGS Comparison: Normalized Area and Compactness

## 11.1 Closing the todobox of Chapter 1

Section 1.3 cited Kmoch et al. (2022)'s qualitative conclusion — that the ISEA-based schemes (rHEALPix, OpenEAGGR, DGGRID) offer the strongest area-preservation guarantee, that H3 has the smallest shape distortion, and that H3 and S2 have the largest cell-area variation — without stating any numeric value from the paper, flagging the reproduction of those values as an open item. This chapter closes that item using the authors' own released data supplement rather than re-deriving the cell geometries independently.

**Methodological Note**

**What was done, precisely.** Kmoch et al. (2022) states that its underlying data are available in open-access repositories, citing "Kmoch & Vasilyev, 2022a, 2022b". The first of these, Kmoch & Vasilyev (2022) (Zenodo, DOI `10.5281/zenodo.6634479`, CC BY 4.0), was located and fetched directly. It contains, for each of the five DGGS and each resolution level the paper studies, a per-resolution summary-statistics file (e.g. `h3_2_stats.csv`) with precomputed `area_min/max/std/mean/median`, `ipq_min/max/std/mean/median` (isoperimetric quotient, a compactness measure with ipq $= 1$ for a circle, $\approx 0.9069$ for a regular hexagon, and $\pi/4 \approx 0.7854$ for a square), `norm_area_min/max/std/mean/median` (cell area normalized so the mean is exactly 1), and `num_cells`. Ten such files were fetched directly (five DGGS $\times$ two resolution levels, 2 and 4) and are tabulated below verbatim, with no value altered, interpolated, or estimated.
**What this is not.** This is a transcription of the authors' own precomputed summary statistics, not an independent recomputation from raw cell polygons: this volume did not regenerate the DGGS tessellations or recompute area and perimeter from scratch, and cannot yet independently confirm the authors' geometric pipeline. That stronger, fully independent reproduction remains open and is listed in Section 11.5.

## 11.2 Normalized area: how uniform are the cells?

Table 11.1 reports the normalized-area statistics at the two resolution levels common to all five implementations in the data supplement.

**Key Result**

Table 11.1 confirms Kmoch et al. (2022)'s qualitative ranking with actual figures: at both resolutions, the normalized-area standard deviation of the three ISEA-based schemes (rHEALPix, DGGRID, OpenEAGGR) sits between roughly 0.002 and 0.021, an order of

Table 11.1: Normalized cell area (mean is 1 by construction) at resolution levels 2 and 4, for all five DGGS studied by Kmoch et al. (2022). Source: Kmoch & Vasilyev (2022), `{h3,s2,rhpix,DGGRID_ISEA7H,eaggr_ISEA3H}_{2,4}_stats.csv`. DGGRID is represented by its ISEA7H configuration (hexagonal, aperture 7, matching H3's cell shape) and OpenEAGGR by its ISEA3H configuration (hexagonal, aperture 3); both are ISEA-based, as the paper's conclusion concerns the ISEA family collectively.

| DGGS | res | $\sigma$(norm. area) | min | max | #cells |
|---|---|---|---|---|---|
| H3 | 2 | 0.13007 | 0.6514 | 1.2051 | 5,736 |
| H3 | 4 | 0.13058 | 0.6472 | 1.2050 | 284,649 |
| S2 | 2 | 0.07520 | 0.9324 | 1.1497 | 85 |
| S2 | 4 | 0.14853 | 0.6936 | 1.2449 | 1,487 |
| rHEALPix | 2 | 0.00193 | 0.9976 | 1.0049 | 453 |
| rHEALPix | 4 | 0.00201 | 0.9975 | 1.0045 | 38,731 |
| DGGRID (ISEA7H) | 2 | 0.02090 | 0.8527 | 1.0140 | 462 |
| DGGRID (ISEA7H) | 4 | 0.00398 | 0.9950 | 1.0090 | 23,608 |
| OpenEAGGR (ISEA3H) | 2 | 0.00735 | 0.9606 | 1.0046 | 862 |
| OpenEAGGR (ISEA3H) | 4 | 0.00262 | 0.9636 | 1.0038 | 45,589 |

magnitude or more below H3's $\approx 0.130$ and S2's $0.075$–$0.149$ — the "strongest guarantee of equal-area preservation" claim is not merely qualitatively true but true by a wide, and now quantified, margin.

*Remark* 11.1 (An observation the two-resolution comparison adds). Fetching two resolution levels, rather than one, surfaces a distinction the qualitative claim alone does not: H3's normalized-area spread is essentially unchanged between resolution 2 and resolution 4 ($0.13007 \to 0.13058$), while S2's grows, roughly doubling ($0.07520 \to 0.14853$). On this evidence alone — two resolution levels for one DGGS pairing — it is not possible to say whether H3's area non-uniformity is a fixed structural property of its construction (plausible, since it is tied to the icosahedron's fixed pentagonal defects, present at every resolution) while S2's grows with resolution, or whether this is specific to the levels sampled; this is flagged as an observation, not a general asymptotic claim, and is listed as an open item below.

## 11.3 Compactness: how close to regular is the average cell?

**Key Result**

Table 11.2 confirms the second qualitative claim of Kmoch et al. (2022): H3's mean ipq ($\approx 0.9047$) is the closest to the ideal regular-hexagon value ($\approx 0.9069$) of any scheme in the table, and remains essentially unchanged across resolution levels, matching "H3 has the smallest shape distortion". The two square/quad-based schemes (S2, rHEALPix) sit near the ideal square value ($\pi/4 \approx 0.7854$), as expected from their cell shape, and are not directly comparable in absolute ipq terms to the hexagon-based schemes; the comparison that matters for shape *distortion* is each scheme's spread relative to its own shape's ideal, not the absolute ipq value, and by that measure H3's tight $\sigma(\text{ipq}) \approx 0.003$ is again the smallest of the three hexagon-based schemes.

Table 11.2: Isoperimetric quotient (compactness; $1 =$ circle, $\approx 0.9069 =$ regular hexagon, $\pi/4 \approx 0.7854 =$ square) at resolution levels 2 and 4. Source: Kmoch & Vasilyev (2022), same files as Table 11.1.

| DGGS | res | mean ipq | $\sigma$(ipq) | cell shape |
|---|---|---|---|---|
| H3 | 2 | 0.90470 | 0.00287 | hexagon/pentagon |
| H3 | 4 | 0.90472 | 0.00280 | hexagon/pentagon |
| S2 | 2 | 0.76741 | 0.01376 | quad |
| S2 | 4 | 0.76709 | 0.01780 | quad |
| rHEALPix | 2 | 0.75693 | 0.04257 | quad (+ dart/skew) |
| rHEALPix | 4 | 0.75394 | 0.04798 | quad (+ dart/skew) |
| DGGRID (ISEA7H) | 2 | 0.89600 | 0.00661 | hexagon/pentagon |
| DGGRID (ISEA7H) | 4 | 0.89638 | 0.00671 | hexagon/pentagon |
| OpenEAGGR (ISEA3H) | 2 | 0.87662 | 0.04243 | hexagon/pentagon |
| OpenEAGGR (ISEA3H) | 4 | 0.87116 | 0.05993 | hexagon/pentagon |

## 11.4 Cross-check: does the reproduced data pass an independent sanity test?

Before treating Tables 11.1 and 11.2 as reliable, two sanity checks were run against the fetched values themselves, independent of any claim in the paper:

- The mean regular-hexagon ipq is $\frac{\pi}{3\sqrt{3}} \approx 0.9069$ (a closed-form constant, not from the paper). All three hexagon-based schemes' mean ipq (H3 0.9047, DGGRID-ISEA7H 0.8960–0.8964, OpenEAGGR-ISEA3H 0.871–0.877) sit just below this ceiling, as required geometrically (no tessellation of a sphere can exceed a regular hexagon's compactness on average, since curvature forces some departure from regularity); none exceeds it, which would have been an internal-consistency failure.
- Each file's own `norm_area_mean` column is 1 (up to floating-point rounding) at every resolution and for every DGGS, as required by the definition of normalized area; this was checked directly against the raw fetched values before any figure was copied into Table 11.1.

Both checks pass, which is evidence the fetched files are what they claim to be, but does not by itself establish that the authors' underlying geometric computation (from which these summary statistics were derived) is itself free of error — that computation was not re-run.

## 11.5 What remains open

**To Complete / Verify**

Three items were left open here and are each taken up in a following chapter, though none is fully closed. (i) A fully independent reproduction — not using Kmoch & Vasilyev (2022)'s files at all, but regenerating cell geometry from an independent software package — is attempted for H3 only in Chapter 12, which confirms compactness precisely but leaves an unexplained 58-cell gap in the reported H3 resolution-2 population; S2, rHEALPix, DGGRID, and OpenEAGGR are not attempted. (ii) Remark 11.1's two-point observation is extended across every resolution with a precomputed statistics file in the release (H3

2–4, S2 2–8) in Chapter 13, which finds S2's growth saturates rather than continuing indefinitely — a materially different conclusion from what the two-point comparison alone suggested. (iii) The connection to Chapter 58's closed-form boundary law is addressed in Chapter 61, which argues (with an explicit self-similarity sketch, not a proof) that real DGGS are resolution-*invariant* rather than boundary-decaying, precisely because recursive subdivision is not the graticule's fixed-interval construction.

This chapter's contribution is exactly as bounded as Section 1.3's original todobox asked: exact figures, drawn directly from the paper's own released data supplement rather than invented or estimated, replacing a purely qualitative citation.

Chapter 12

# An Independent Reproduction: H3 Geometry from Scratch

## 12.1 Scope: item (i) of the previous chapter's todobox

Section 11.5 left three items open. This chapter attempts the first and strongest: computing normalized-area and compactness statistics for a real DGGS *entirely independently* of Kmoch et al. (2022); Kmoch & Vasilyev (2022) — not transcribing their precomputed summary statistics (as Chapter 11 did), and not even reprojecting their released raw cell polygons, but regenerating the cell tessellation itself from an independent software package and computing area and perimeter with an independently chosen algorithm.

**Methodological Note**

**What was done, precisely.** H3 resolutions 2 and 3 were regenerated in full (globally, all base cells and their children) using `h3-py` v4.5.0, a Python binding to Uber's own H3 C library — a different codebase from the `dggrid4py/geopandas` pipeline Kmoch et al. (2022) used. Cell area was taken from `h3-py`'s own `cell_area()` (H3's built-in geodesic area engine); cell perimeter was computed independently by this volume, as the sum of great-circle (haversine) distances between consecutive boundary vertices. Compactness ipq $= 4\pi\,\text{area}/\text{perimeter}^2$ and normalized area (area divided by the resolution's own mean) were then computed from these two independently-sourced quantities. Full code: `code/independent_h3_reproduction.py`; full output: `data/independent_h3_reproduction_results.json`.

**A dead end, reported rather than hidden.** The first version of this script computed area itself, from the raw lat/lon boundary, via the closed-form spherical-excess formula (Chamberlain & Duquette, 2007; Green's theorem on the sphere). For ordinary cells this agreed with `h3-py`'s `cell_area()` to several significant figures. For the six H3 cells whose boundary rings the North Pole (e.g. `830326fffffffff` at resolution 3), it did not: the hand-rolled formula returned 2434 $\text{km}^2$ against `h3-py`'s own 13,521 $\text{km}^2$ for the same cell, a factor of 5.5 too small. This is a genuine numerical-instability failure of the spherical-excess sum for a polygon whose interior nearly encircles a pole, not a data error; it was caught only by cross-checking against `h3-py`'s reported `average_hexagon_area(3)`, which the buggy figure violated. The fix was to use `h3-py`'s own `cell_area()` throughout rather than repair the formula. This episode is left in the chapter deliberately: it is a small, concrete instance of exactly the pole-boundary fragility that motivates Chapter 58's distinction between bulk and boundary behaviour, encountered here in an implementation rather than derived in a proof.

## 12.2 Results and comparison to the transcribed table

Table 12.1 reports the independently computed statistics alongside Tables 11.1 and 11.2's transcribed values for the same two resolutions.

Table 12.1: H3 resolutions 2–3: this chapter's from-scratch computation (via `h3-py`, global, all cells including the 12 pentagons) against Kmoch & Vasilyev (2022)'s transcribed values (via `dggrid4py`/LAEA reprojection, a filtered subset). Source: `code/independent_h3_reproduction.py`.

| res | source | $\sigma$(norm. area) | norm. area min | ipq mean | #cells |
|---|---|---|---|---|---|
| 2 | this chapter (independent) | 0.13373 | 0.5181 | 0.90481 | 5,882 |
| 2 | Kmoch & Vasilyev (2022) (transcribed) | 0.13007 | 0.6514 | 0.90470 | 5,736 |
| 3 | this chapter (independent) | 0.13346 | 0.5097 | 0.90471 | 41,162 |
| 3 | Kmoch & Vasilyev (2022) (transcribed) | 0.13050 | 0.6476 | 0.90473 | 40,534 |

**Key Result**

The mean compactness ipq agrees to four significant figures at both resolutions (0.90481 vs. 0.90470 at res. 2; 0.90471 vs. 0.90473 at res. 3) despite the two computations sharing no code, no cell-generation library, and no area algorithm — this is the strongest form of independent confirmation Chapter 11 could offer for Kmoch et al. (2022)'s "H3 has the smallest shape distortion" claim. The normalized-area *minimum* does not agree as closely (0.518 vs. 0.651 at res. 2): this chapter's global count is exactly the textbook H3 resolution-2 total of 5,882 cells, including all twelve icosahedral pentagon cells, whose normalized area clusters tightly around 0.51 (verified directly in the underlying data: all twelve res.-3 pentagons have normalized area in $[0.5097, 0.5097]$ to four digits). Kmoch & Vasilyev (2022)'s reported count of 5,736 is 146 cells short of the full globe; their own `date_line_cross_error_cells` column accounts for only 88 of that gap at resolution 2, leaving 58 cells unaccounted for by any column in their released summary file.

**To Complete / Verify**

Whether the remaining 58 (res. 2) / 628 (res. 3, $41{,}162 - 40{,}534 - 218$) unaccounted-for cells are the twelve pentagons plus their immediate neighbours, or something else entirely, is not resolved here: it would require the exact filtering logic of Kmoch et al. (2022)'s processing pipeline, documented in Kmoch et al. (2022)'s companion code release (Kmoch & Vasilyev, 2022a, not independently fetched here), rather than the summary statistics alone. Until that is checked, the honest reading of Table 12.1 is: *compactness* is confirmed independently and precisely; the *extreme low end of the area-normalization range* is not directly comparable between the two computations, because it is not established that they are describing the same population of cells.

## 12.3 What remains open

**To Complete / Verify**

This closes item (i) of Section 11.5 only partially and only for H3. S2, rHEALPix, DGGRID, and OpenEAGGR each have their own Python or command-line bindings (`s2sphere`, `rHEALPixDGGS`, the DGGRID CLI, RiskAware's OpenEAGGR library respectively) and could in principle receive the same treatment; none was attempted here. A first exploratory attempt at enumerating S2 cells natively (via `s2sphere`) surfaced a resolution-to-level mapping ambiguity — this volume's naive level-2 enumeration produced 96 cells globally, not Kmoch et al. (2022)'s reported 85 for their "S2 resolution 2" — that was not resolved in the time available and is left as a concrete, well-defined next step rather than papered over with a guessed correspondence.

*Update:* Chapter 14 returns to exactly this question with a full systematic investigation across all seven reported S2 resolutions, substantially narrowing (though not completely closing) the ambiguity.

*Further update:* Chapter 15 gives DGGRID the same from-scratch treatment attempted here for H3, finding the identical pole-boundary spherical-excess failure mode in a second, independently built codebase, and an analogous (though numerically different) unexplained cell-count gap.

*Final update:* Chapter 16 gives OpenEAGGR the same treatment, closing this item for all five schemes. Its spherical-excess-vs.-`pyproj` disagreement is larger, not smaller, than the pole-boundary cases found for H3 and DGGRID, and its own failure mode is different in kind: not a pole-boundary area error but a small number of cells (at the tessellation's extremal row coordinate) whose outline the library itself returns with a literal NaN vertex.

*Source-code update:* Section 14.6 subsequently locates Kmoch et al.'s own S2 processing code (Kmoch & Vasilyev, 2022a) and tests, directly against that code's exact call pattern, whether a region-covering algorithm (as opposed to leaf-level enumeration) explains the S2 deficit reported here; it finds this specific mechanism does not (an independent reimplementation of the same algorithm returns zero deficit), narrowing but not closing the question.

Chapter 13

# Does the H3/S2 Divergence Persist? The Full Resolution Range

## 13.1 Scope: item (ii) of the reproduction chapter's todobox

Remark 11.1 noticed, from two resolution levels alone, that H3's normalized-area spread was essentially flat between resolutions 2 and 4 while S2's roughly doubled, and explicitly declined to call this a trend rather than an artefact of the two levels sampled. This chapter checks it against every resolution for which Kmoch & Vasilyev (2022) released a precomputed summary-statistics file: three levels for H3 (resolutions 2–4; no `_stats.csv` file exists in the release for H3 resolutions 5 or higher, only raw geometry) and seven for S2 (resolutions 2–8; resolution 9 exists in the release but was not independently fetched here).

## 13.2 The extended table

Table 13.1: Normalized-area statistics across every available resolution. Source: Kmoch & Vasilyev (2022), `{h3,s2}_{2..8}_stats.csv`, `data/h3_s2_resolution_extension_results.json`.

| DGGS / res | $\sigma$(norm. area) | min | max | #cells |
|---|---|---|---|---|
| H3 2 | 0.13007 | 0.6514 | 1.2051 | 5,736 |
| H3 3 | 0.13050 | 0.6476 | 1.2049 | 40,534 |
| H3 4 | 0.13058 | 0.6472 | 1.2050 | 284,649 |
| S2 2 | 0.07520 | 0.9324 | 1.1497 | 85 |
| S2 3 | 0.13738 | 0.7944 | 1.2474 | 364 |
| S2 4 | 0.14853 | 0.6936 | 1.2449 | 1,487 |
| S2 5 | 0.15083 | 0.6488 | 1.2584 | 6,017 |
| S2 6 | 0.15148 | 0.6540 | 1.2566 | 24,201 |
| S2 7 | 0.15167 | 0.6519 | 1.2577 | 97,065 |
| S2 8 | 0.15172 | 0.6527 | 1.2576 | 388,775 |

**Key Result**

The remark's flat/growing distinction survives, but its character changes qualitatively once more resolutions are in view. H3's $\sigma$ does not merely stay flat between two points; it stays flat across all three available points ($0.13007 \to 0.13050 \to 0.13058$, a total drift of under $4\%$ over two full resolution doublings). S2's $\sigma$ does not grow indefinitely, as "roughly doubling" from two points alone might have suggested: it rises sharply from resolution 2 to 3 ($0.07520 \to 0.13738$, a factor of $1.83$), continues rising more slowly through resolutions 4–5, and then *saturates* — resolutions 6, 7, and 8 give $0.15148$, $0.15167$, $0.15172$, differing in

the fourth decimal place. S2's normalized-area spread is converging to a fixed asymptotic value near $0.152$, not diverging.

*Remark* 13.1 (The minimum is noisier than the standard deviation). S2's normalized-area *minimum* does not converge as cleanly as its standard deviation: $0.9324, 0.7944, 0.6936, 0.6488, 0.6540, 0.6519, 0.6527$ drops sharply through resolution 5 and then *oscillates* in the fourth digit around $\approx 0.652$ rather than settling monotonically, unlike $\sigma$, which decreases its increments monotonically at every step. This is expected on general statistical grounds: $\sigma$ is a bulk statistic, an average over tens of thousands to hundreds of thousands of cells, and averages have shrinking sampling variability; the minimum is an extreme-value statistic, determined by whichever single most-distorted cell happens to exist at that resolution's specific tessellation, and inherits none of that averaging stability. The same bulk/extreme distinction structures Chapter 58's entire argument about the toy graticule; here it resurfaces empirically, in the noise pattern of a real dataset, rather than as a proved pair of convergence rates. The connection is made precise in Chapter 61.

## 13.3 What remains open

**To Complete / Verify**

This closes item (ii) of Section 11.5 for H3 and S2 only; rHEALPix, DGGRID, and OpenEAGGR each also have precomputed statistics files across resolutions 2–5 or 2–6 in the same release and were not tabulated here. S2 resolution 9, and any explanation of why S2's normalized-area spread saturates specifically near $0.152$ rather than some other value (a question about S2's quadrilateral cube-face projection this volume does not attempt), are both left open.

Chapter 14

# The S2 Resolution-to-Level Ambiguity, Systematically

## 14.1 Scope: returning to a question left open

Section 12.3 left a loose end: a first, exploratory native enumeration of S2 cells (via `s2sphere`, independently of whatever S2 binding Kmoch et al. (2022) used) produced 96 cells globally at S2 level 2, not the 85 cells Kmoch et al. (2022) report for their "S2 resolution 2". That chapter declined to guess at a correspondence and left the question open. This chapter returns to it with the full resolution range Chapter 13 already tabulated (S2 resolutions 2–8) and tests two candidate mechanisms for the deficit, rather than one isolated data point.

**Methodological Note**

**What was done, precisely.** For each S2 level $L = 2, \ldots, 8$, every cell at that level was enumerated natively via `s2sphere` v0.2.5 (`CellId.begin(L)` stepped with `.next()` to `CellId.end(L)`), giving the textbook global count $6 \cdot 4^L$ with no filtering of any kind. For each cell, the four boundary vertices were converted to (lat, lon) via `LatLng.from_point`, and the cell was flagged as *date-line-crossing* if any consecutive pair of boundary vertices differs in longitude by more than $180°$ — the same test this volume already applies to H3 cells in `independent_h3_reproduction.py`. The cell center's absolute latitude was also recorded. Full code: `code/s2_resolution_ambiguity.py`; full output: `data/s2_resolution_ambiguity_results.json`.

## 14.2 Mechanism one: date-line-crossing removal

This is the parameter-free candidate: no threshold is chosen, only a geometric test already used elsewhere in this volume, and it is externally corroborated in the sense that Kmoch et al. (2022)'s own H3 release ships a `date_line_cross_error_cells` column for the same phenomenon (Chapter 12), so a resolution-dependent count of cells excluded for crossing the anti-meridian is not an invented category.

*Remark* 14.1 (The date-line count has an exact closed form). At every one of the seven levels tested, the date-line-crossing count equals $2^{L+1}+1$ exactly: $9 = 2^3+1$, $17 = 2^4+1$, $33 = 2^5+1$, $65 = 2^6+1$, $129 = 2^7+1$, $257 = 2^8+1$, $513 = 2^9+1$. This is reported as a verified empirical pattern across the range tested, not as a proved general formula: no derivation from S2's face structure is attempted here, and whether it continues to hold at levels beyond 8 is not checked.

**Key Result**

Date-line-crossing removal is real, exactly quantifiable, and explains a non-trivial share of the deficit at every level tested — but that share is not stable. It peaks near resolution 3

Table 14.1: S2 levels 2–8: naive global count, Kmoch et al. (2022)'s reported count, the resulting deficit, and the date-line-crossing count at each level. Source: `data/s2_resolution_ambiguity_results.json`.

| level | naive $6 \cdot 4^L$ | Kmoch et al. (2022) | deficit | date-line count | deficit share |
|---|---|---|---|---|---|
| 2 | 96 | 85 | 11 | 9 | 81.8% |
| 3 | 384 | 364 | 20 | 17 | 85.0% |
| 4 | 1,536 | 1,487 | 49 | 33 | 67.3% |
| 5 | 6,144 | 6,017 | 127 | 65 | 51.2% |
| 6 | 24,576 | 24,201 | 375 | 129 | 34.4% |
| 7 | 98,304 | 97,065 | 1,239 | 257 | 20.7% |
| 8 | 393,216 | 388,775 | 4,441 | 513 | 11.6% |

(85.0% of the deficit) and then falls steadily to barely a ninth of the deficit by resolution 8 (11.6%). Over the six level-steps from resolution 2 to resolution 8, the date-line count grows by a factor of 57 ($9 \to 513$, consistent with its closed form $2^{L+1}+1$ roughly doubling each level), while the deficit itself grows by a factor of 404 ($11 \to 4{,}441$) over the same span — faster than a count that doubles each level, but slower than the factor of 4,096 ($= 4^6$) a purely area-proportional count would give. Whatever mechanism accounts for the remainder therefore scales faster than the one-dimensional line of date-line-crossing cells, but not as fast as a fixed-area polar cap would, which is exactly what motivates testing (and then having to qualify) the second candidate below.

## 14.3 Mechanism two: a reverse-fit polar-cap threshold

The second candidate is explicitly *not* derived from any property of the S2 projection. It is a reverse fit: for each level, this volume searched for a single latitude cutoff $L^*$ such that the *union* of the date-line-crossing cells and the cells whose center lies at $|\text{lat}| \geq L^*$ has size exactly equal to the deficit. The search is an $O(n \log n)$ sweep from the pole inward over the distinct tied latitude values that actually occur in the tessellation (code: `find_union_threshold` in `code/s2_resolution_ambiguity.py`), so it either lands on the deficit exactly or it does not; there is no interpolation and no rounding to the nearest available count.

Table 14.2: Outcome of the reverse-fit search at each level. "Exact match" gives the latitude cutoff found; "no exact threshold" reports the two union-counts the sweep jumps between at the tie-block that straddles the target deficit. Source: `data/s2_resolution_ambiguity_results.json`.

| level | deficit | outcome | detail |
|---|---|---|---|
| 2 | 11 | exact match | $\|\text{lat}\| \geq 75.149°$ |
| 3 | 20 | no exact threshold | jumps $19 \to 26$ at $\|\text{lat}\| = 72.787°$ |
| 4 | 49 | no exact threshold | jumps $46 \to 53$ at $\|\text{lat}\| = 76.251°$ |
| 5 | 127 | exact match | $\|\text{lat}\| \geq 77.871°$ |
| 6 | 375 | exact match | $\|\text{lat}\| \geq 78.262°$ |
| 7 | 1,239 | no exact threshold | jumps $1{,}234 \to 1{,}242$ at $\|\text{lat}\| = 78.260°$ |
| 8 | 4,441 | no exact threshold | jumps $4{,}437 \to 4{,}445$ at $\|\text{lat}\| = 78.319°$ |

**Key Result**

The reverse fit succeeds at exactly three of the seven levels tested (2, 5, 6) and fails at the other four (3, 4, 7, 8), and the failures are not near misses: at each failing level the sweep skips directly over the target deficit because a whole tied block of cells — all sharing the same center latitude to the precision `s2sphere` returns — crosses the pole-ward boundary at once. A single-latitude cutoff on cell-center latitude is, on this evidence, *not* the mechanism Kmoch et al. (2022) used: if it were, the search would succeed at every level, not three of seven. The three successes are better read as coincidences of the tie structure at those particular resolutions than as confirmation of a latitude-circle rule.

## 14.4 Follow-up: face-relative coordinates instead of latitude

The most natural next step, once a plain latitude cutoff had failed at four of seven levels, is to ask whether a cutoff defined in S2's own cube-face coordinates – rather than in geographic latitude – might do better. A radius natural to the quadrilateral face grid need not trace a clean latitude circle once mapped back onto the sphere, so this is a genuinely different candidate, not a relabelling of the first. It was checked directly rather than left as a suggestion.

**Methodological Note**

**What was done, precisely.** Only S2 faces 2 (north pole) and 5 (south pole) can contain the cells in question at all: an exhaustive check at level 8 confirmed that the four equatorial faces never exceed $|\text{lat}| = 44.85°$, far below every latitude threshold found in Section 14.3. For each cell on faces 2 and 5, this volume computed its center's $(u, v)$ coordinate in that face's own square (via `CellId.get_center_uv()`, where $(0, 0)$ is the face center, i.e. the pole itself) and its Chebyshev radius $\max(|u|, |v|)$ — the natural "distance from the pole" in the quadrilateral cube-face grid, as opposed to angular distance on the sphere. The same union-with-date-line-crossing sweep as Section 14.3 was then repeated, swept outward in $(u, v)$-radius from the pole instead of in latitude. Full code: `run_uv_followup` in `code/s2_resolution_ambiguity.py`; full output: `data/s2_resolution_ambiguity_uv_followup_results.json`.

Table 14.3: Outcome of the face-relative $(u, v)$-radius reverse-fit search at each level. Every level overshoots the deficit at the first tie-block encountered — no level achieves an exact match, worse than the latitude-based search's three successes out of seven. Source: `data/s2_resolution_ambiguity_uv_followup_results.json`.

| level | deficit | outcome | overshoot at tie-block |
|---|---|---|---|
| 2 | 11 | no exact threshold | 14 at uv-radius 0.1875 |
| 3 | 20 | no exact threshold | 22 at uv-radius 0.0885 |
| 4 | 49 | no exact threshold | 60 at uv-radius 0.1367 |
| 5 | 127 | no exact threshold | 130 at uv-radius 0.1123 |
| 6 | 375 | no exact threshold | 404 at uv-radius 0.1244 |
| 7 | 1,239 | no exact threshold | 1,384 at uv-radius 0.1306 |
| 8 | 4,441 | no exact threshold | 4,698 at uv-radius 0.1275 |

**Key Result**

The face-relative $(u, v)$-radius candidate is not an improvement over plain geographic latitude; it is worse. It fails to hit the deficit exactly at every one of the seven levels tested, where the latitude-based cutoff at least succeeded at three. This rules out — rather than confirms — the hypothesis that the filtering follows a simple radius in the cube-face's own quadrilateral coordinates. Two candidate "natural" geometric cutoffs (a latitude circle, and a face-square Chebyshev radius) have now both been tested directly and both rejected as the whole explanation; whatever Kmoch et al. (2022) used, on this evidence, is not a simple one-parameter cutoff on either geographic or cube-face coordinates alone.

## 14.5 What this narrows, and what it still leaves open

## 14.6 A third candidate: Kmoch et al.'s own processing code, located and tested

The previous section's todobox named one path not yet attempted: obtaining Kmoch et al. (2022)'s own processing code rather than continuing to reverse-engineer their filtering criterion from summary counts alone. This revision locates it. The paper's own data-availability statement cites a code supplement (Kmoch & Vasilyev, 2022a), archived at Zenodo and mirrored on GitHub at `LandscapeGeoinformatics/dggs_comparisons`; its `s2_helper.py` module is fetched and read directly here for the first time in this volume.

**Methodological Note**

**What the source code actually does.** Kmoch et al.'s `get_s2_cells(res, extent=None)` function does *not* enumerate S2 cells natively via `CellId` iteration, as this volume's own reproduction (Section 14.1) does. It instead constructs an `S2RegionCoverer`, fixes both `min_level` and `max_level` to the target resolution, builds a whole-sphere `S2LatLngRect` from $(-90°, -180°)$ to $(90°, 180°)$, and calls `GetCovering` on that rectangle – a *region-covering* algorithm, designed to approximate an arbitrary region with a reasonably small set of cells, rather than a guaranteed enumeration of every cell of a given level. This is a substantively different algorithm from this volume's own leaf-level iteration, and a plausible-looking candidate mechanism for the deficit: region-covering algorithms are permitted to behave unpredictably at degenerate inputs, and a rectangle spanning the entire sphere, including both poles, is about as degenerate an input as an axis-aligned lat/lng rectangle can receive.

This candidate is directly testable, and was tested rather than left as a plausible-sounding conjecture. Using `s2sphere`'s own `RegionCoverer` class (v0.2.5, the same codebase already used throughout this chapter, though not the same S2 binding Kmoch et al. used, which was Google's official `pywraps2` C++ bindings, not installable in this environment without a from-source build outside this volume's network access), this revision reproduces Kmoch et al.'s exact call sequence – `min_level == max_level == L`, whole-sphere `LatLngRect` – at all seven levels $L = 2, \ldots, 8$ (full code: `code/s2_regioncoverer_mechanism_test.py`; full output: `data/s2_regioncoverer_mechanism_test_results.json`).

**Key Result**

The region-covering call returns *exactly* $6 \cdot 4^L$ cells at every one of the seven levels tested – identical to this volume's own naive leaf-level enumeration, with zero deficit at any level, and with every returned cell reported at the requested level (no coarser parent cells substituted anywhere). This rules out, rather than confirms, the region-covering-algorithm-vs-leaf-enumeration distinction as the mechanism behind Kmoch et al.'s deficit: at least in `s2sphere`'s implementation of the same algorithm, `RegionCoverer.get_covering` against a whole-sphere rectangle with a fixed level is exactly equivalent to full leaf-level enumeration, with no approximation loss at the poles or anywhere else.

This is a genuine methodological advance even though the mechanism it tests fails: this chapter no longer treats Kmoch et al.'s processing pipeline as an unknown black box producing an unexplained count. The exact function and call pattern that produced their reported counts are now known and cited (Section 14.6's disclaimerbox), and one specific, well-motivated hypothesis about how that pattern could explain the deficit has been tested directly and eliminated. What remains genuinely open is narrower than before: since the region-covering algorithm itself is not the culprit (at least as implemented in `s2sphere`), the residual explanation must lie either in a behavioral difference between `s2sphere`'s `RegionCoverer` and Google's own `pywraps2` implementation of the identical algorithm (not tested here, and not testable without building the C++ library), or in some other stage of Kmoch et al.'s pipeline entirely (e.g. a post-hoc filter applied after `GetCovering` returns, not visible in `get_s2_cells` alone but possibly present in the notebook code that calls it, which was not fetched in this pass).

**To Complete / Verify**

**Narrowed further, still not closed.** Three candidate mechanisms have now been tested and rejected: a latitude-circle cutoff (three of seven levels only), a face-relative $(u, v)$-radius cutoff (zero of seven), and, newly, the region-covering-vs-leaf-enumeration algorithm distinction itself (zero deficit found, i.e. no distinction at all in `s2sphere`'s implementation). Kmoch et al.'s own source code is now known and cited rather than merely inferred to exist, which is real progress in the source-critical sense — but the specific C++ S2 library binding they used (`pywraps2`) was not installable in this environment, so their exact numerical behavior remains unverified rather than merely unexplained. The next concrete step, not yet attempted, would be to build `pywraps2` from source (or find an environment where it is already available) and run Kmoch et al.'s own `get_s2_cells` function verbatim, rather than a same-algorithm reimplementation in a different codebase, to check whether the deficit is a genuine behavioral difference between S2 implementations or arises elsewhere in their pipeline (for instance, a filtering step outside `s2_helper.py` itself, in code this pass did not fetch).

Chapter 15

# A Second Independent Reproduction: DGGRID Geometry from Scratch

## 15.1 Scope: the DGGRID half of the previous chapter's open item

Section 12.3 closed Section 11.5's strongest reproduction item for H3 only, naming DGGRID (among others) as untouched. This chapter takes up DGGRID specifically, in the same spirit: not transcribing Kmoch & Vasilyev (2022)'s precomputed summary statistics (as Chapter 11 did for all five schemes), but regenerating the cell tessellation itself from an independently built copy of the reference software and computing area and compactness with independently chosen methods.

**Methodological Note**

**What was done, precisely.** Sahr (2026)'s DGGRID source (v9.03b, commit `04bf5ed`) was cloned from its GitHub repository and built from scratch with `cmake/g++` (`WITH_GDAL=OFF`; GDAL is not required for the plain-GeoJSON output path used here) — a from-source build by this volume, not a precomputed binary or a third-party wrapper such as `dggrid4py`. `GENERATE_GRID` was run for the `ISEA7H` topology (hexagonal, aperture 7, the same configuration Kmoch et al. (2022) used for DGGRID) over the whole globe at resolutions 2 and 4, emitting every cell's WGS84 lat/lon boundary as GeoJSON polygons. Area and perimeter were then computed by two independent methods on that output: (a) the same closed-form spherical-excess formula (Chamberlain & Duquette, 2007) used for H3's perimeter in Chapter 12, with great-circle haversine edge lengths; (b) `pyproj`'s `Geod.geometry_area_perimeter`, a separately implemented, WGS84-ellipsoid geodesic engine. Both quantities of interest, normalized area and ipq, are scale- and sphere-vs-ellipsoid-invariant by construction, so the two methods should agree closely; any persistent gap is reported, not smoothed over. Full code: `code/dggrid_independent_reproduction.py`; full output: `data/dggrid_independent_reproduction_results.json`.

**The same dead end, recurring in a different codebase.** Method (a) was tried first, exactly as in Chapter 12. For ordinary cells it agreed with method (b) to three or four significant figures. For a small number of cells whose boundary lies within a few degrees of a pole (e.g. cell 6 at resolution 2, spanning longitudes from $-79°$ to $101°$ at latitudes 80–87°N), it did not: method (a) returned a normalized area of 1.542 for that cell against method (b)'s 1.007, a factor of roughly 1.5 too large — the same class of failure documented for H3's own polar cells in Chapter 12 (there a $5.5\times$ *under*-estimate; here a $\sim 1.5\times$ *over*-estimate, so the direction of the error is not itself stable, only its location). This is now observed in a second DGGS, built by a second team, generated by a third piece of software, computed by this volume's own from-scratch spherical-excess code in both cases: the evidence that the Chamberlain–Duquette sum is specifically pole-boundary-fragile, rather than an H3- or implementation-specific quirk, is correspondingly stronger. As before, the fix is to prefer

the more robust method (here, `pyproj`'s ellipsoidal geodesic engine) rather than repair the formula; all results below use method (b).

## 15.2 Results and comparison to the transcribed table

Table 15.1 reports the independently computed statistics alongside Tables 11.1 and 11.2's transcribed DGGRID (ISEA7H) values for the same two resolutions.

Table 15.1: DGGRID ISEA7H resolutions 2 and 4: this chapter's from-scratch computation (built from source, global, all cells including the 12 pentagons, area/perimeter via `pyproj` WGS84 geodesics) against Kmoch & Vasilyev (2022)'s transcribed values (via their own `dggrid4py`/LAEA reprojection pipeline). Source: `code/dggrid_independent_reproduction.py`.

| res | source | $\sigma$(norm. area) | norm. area min | ipq mean | $\sigma$(ipq) | #cells |
|---|---|---|---|---|---|---|
| 2 | this chapter (independent) | 0.02335 | 0.8531 | 0.89837 | 0.00698 | 492 |
| 2 | Kmoch & Vasilyev (2022) (transcribed) | 0.02090 | 0.8527 | 0.89600 | 0.00661 | 462 |
| 4 | this chapter (independent) | 0.00507 | 0.8602 | 0.89643 | 0.00674 | 24,012 |
| 4 | Kmoch & Vasilyev (2022) (transcribed) | 0.00398 | 0.9950 | 0.89638 | 0.00671 | 23,608 |

**Key Result**

As with H3, mean compactness agrees closely despite the two computations sharing no code, no cell-generation library, and (at resolution 4) no area algorithm: 0.89837 vs. 0.89600 at resolution 2 (0.3% relative difference), 0.89643 vs. 0.89638 at resolution 4 (agreement to four significant figures). This chapter's independently generated tessellation reproduces exactly twelve pentagon cells at both resolutions — the icosahedron's fixed vertex defects, structurally the same feature already identified for H3's own pentagons in Chapter 12 — and these twelve cells are, in this independent computation, unambiguously the smallest-area cells in the tessellation (normalized area 0.8531–0.8602 at both resolutions checked; every hexagon at resolution 2 has normalized area in $[0.9881, 1.0148]$, confirmed directly in the underlying per-cell data). Kmoch & Vasilyev (2022)'s reported resolution-4 minimum of 0.9950 is therefore not the pentagon population — either their pipeline's normalized-area minimum excludes the pentagons entirely, or its area computation assigns pentagons a value close to the hexagon range, neither of which can be determined from the summary statistics alone.

**To Complete / Verify**

Two gaps remain unexplained, in the same spirit as Chapter 12's unresolved 58-cell gap for H3. **(a) Cell count.** This chapter's global `ISEA7H` generation, using DGGRID's own deterministic aperture-7 growth law $10 \cdot 7^n + 2$ cells at DGGRID resolution index $n$ (verified directly: $12, 72, 492, 3432, 24{,}012$ for $n = 0, \ldots, 4$, exactly matching this formula at every level tested), gives 492 cells at the resolution this chapter treats as "2" and 24,012 at "4", against Kmoch & Vasilyev's reported 462 and 23,608 — gaps of 30 and 404 cells respec-

tively, neither a multiple of the other and neither equal to twice the pentagon count (24) or any other obvious structural quantity. Since DGGRID resolutions and Kmoch et al.'s own resolution labels are not shown anywhere in the paper's public abstract-level material to be the same index space (an ambiguity of exactly the kind Chapter 14 investigated for S2), the two resolution-"2" and resolution-"4" rows being compared here are not positively confirmed to describe the same DGGRID configuration, only the same nominal ISEA7H topology at plausible-looking cell counts. **(b) The resolution-4 minimum.** Independent of (a), the specific disagreement at the low end of resolution 4 (0.860 here vs. 0.995 transcribed) is large enough that it cannot be dismissed as a rounding or precision artifact; it is left open rather than resolved by assumption. Both items would require Kmoch et al.'s own processing pipeline (Kmoch & Vasilyev, 2022a, not independently fetched here, as already noted for the analogous H3 gap) to run down further.

## 15.3 What remains open

**To Complete / Verify**

This closes the DGGRID portion of Section 12.3's item (i); OpenEAGGR (RiskAware's ISEA3H implementation) remains entirely untouched by any from-scratch reproduction in this volume, and is the last of the five schemes Kmoch et al. (2022) studied for which that is true. rHEALPix, by contrast, has since received extensive independent, from-scratch treatment of its own in Chapters 97 to 101, though aimed at real-tradeoff and operational-timing questions rather than at this specific two-resolution area/compactness comparison; no attempt is made here to retrofit that separate line of work into Table 15.1's exact format.

Chapter 16

# A Third Independent Reproduction: OpenEAGGR Geometry from Scratch

## 16.1 Scope: the last of the five schemes

Section 15.3 closed the DGGRID half of Section 12.3's reproduction item and named OpenEAGGR (RiskAware's ISEA3H implementation) as the last of Kmoch et al. (2022)'s five schemes still untouched by any from-scratch reproduction in this volume. This chapter closes that item, in the same spirit as Chapter 12 for H3 and Chapter 15 for DGGRID: not transcribing Kmoch & Vasilyev (2022)'s precomputed summary statistics (already tabulated in Chapter 11), but regenerating the cell tessellation itself from an independently built copy of the reference software and computing area and compactness with independently chosen methods.

**Methodological Note**

**What was done, precisely.** RiskAware (2017)'s OpenEAGGR source was cloned from its GitHub repository (commit `c0d6d3b`) and built from scratch: all 35 C++ source files (32 under `EAGGR/Src`, 3 under `EAGGR/API`) were compiled with `g++` against the repository's own bundled GDAL and proj4 headers and static libraries (`libgdal.a`, `libproj.a`, both included in `EAGGR/3rdParty`) and linked into a single shared library, `libeaggr.so` — a from-source build by this volume, not a precomputed binary. The official Python wheel was not used, since it expects separately built GDAL/proj shared objects; instead a minimal `ctypes` wrapper (`code/eaggr_wrap.py`) calls `libeaggr.so` directly. Version confirmed at runtime: `EAGGR_GetVersion()` → `"v2.0"`.

**Two library bugs encountered and disclosed, not silently worked around.** (a) Calling `EAGGR_DeallocateString` on the string returned by `EAGGR_ConvertDggsCellOutlineToShapeString` segfaults this build, regardless of resolution; every outline-extraction call in this chapter's code therefore reads the returned string and does not free it, a disclosed, deliberate small memory leak acceptable for a one-off measurement run, not a production usage pattern. (b) At resolution 4, exactly 80 of 7,580 enumerated cells — precisely those sitting at the extremal row-coordinate value ($-16$, the tessellation's minimum at this resolution) in OpenEAGGR's internal offset coordinates — have one vertex of their WKT outline returned as the literal string `nan nan` by the library itself (verified by printing the raw WKT before any parsing by this volume's code, so the fault is inside `libeaggr.so`, not in the parsing). This pattern (exactly the extremal-row cells, no others) is consistent with a known class of DGGS artifact: degenerate outlines at the twelve fixed singular vertices of the icosahedron, where hexagonal offset coordinates cannot be extended smoothly. These 80 cells are excluded from the area/compactness statistics below and counted explicitly as `skipped_cells`; no coordinate is invented, interpolated, or silently repaired to replace them.

**Cell enumeration is not a tree.** RiskAware (2017)'s own Python wrapper declares `_EAGGR_MAX_PARENT_CELLS = 3`, confirmed directly: `EAGGR_GetDggsCellParents` does return three distinct parents for some cells, so recursive child enumeration from the 20 icosahedron-face base cells (resolution 0) requires de-duplication by cell-ID string at every level, exactly as Section 16.2 below documents; without it, cell counts are overcounted starting at resolution 2.
**Area and perimeter** were computed by the same two independent methods used for H3 and DGGRID: (a) the closed-form spherical-excess formula (Chamberlain & Duquette, 2007) with great-circle haversine edge lengths; (b) `pyproj`'s `Geod.geometry_area_perimeter`, a separately implemented, WGS84-ellipsoid geodesic engine. Full code: `code/eaggr_independent_reproduction.py`, `code/eaggr_growth_factor.py`; full output: `data/eaggr_independent_reproduction_results.json`, `data/eaggr_isea3h_growth_results.json`.

## 16.2 The growth factor: from aperture 7 at resolution 1 to aperture 3 in the limit

Before any area computation, the cell-count growth rate itself needed establishing, since (unlike DGGRID's deterministic $10 \cdot 7^n + 2$ law, confirmed directly in Chapter 15) OpenEAGGR's ISEA3H offset grid has no such published closed form available within this volume, and the parent-child relation is not a tree. Starting from all 20 base cells and recursively calling `EAGGR_GetDggsCellChildren` with de-duplication by cell-ID string at each level gives Table 16.1.

Table 16.1: Deduplicated ISEA3H cell counts by resolution, all 20 base faces, this chapter's own enumeration. "Raw" is the un-deduplicated sum of children reported at the previous level (i.e. what naive recursive enumeration without de-duplication would count). Source: `code/eaggr_growth_factor.py`.

| resolution | raw (undeduped) | deduped cells | growth (deduped/prev) |
|---|---|---|---|
| 0 | – | 20 | – |
| 1 | 140 | 140 | 7.000 |
| 2 | 980 | 560 | 4.000 |
| 3 | 3,920 | 2,220 | 3.964 |
| 4 | 15,540 | 7,580 | 3.414 |
| 5 | 53,060 | 24,860 | 3.280 |
| 6 | 174,020 | 78,620 | 3.163 |
| 7 | 550,340 | 243,980 | 3.103 |
| 8 | 1,707,860 | 746,460 | 3.060 |
| 9 | 5,225,220 | 2,267,020 | 3.037 |

**Key Result**

The first refinement step (resolution $0 \rightarrow 1$) is a clean aperture-7 subdivision with no overcounting at all (140 raw $=$ 140 deduped, matching $20 \times 7$ exactly): every base cell's 7 children are distinct from every other base cell's children. From resolution 1 onward, de-duplication removes a rapidly shrinking fraction of the raw count, and the deduped growth factor falls monotonically from 7.0 toward 3.0:

4.000, 3.964, 3.414, 3.280, 3.163, 3.103, 3.060, 3.037. This is exactly the asymptotic behaviour expected of an aperture-3 hexagonal grid (the "3H" of ISEA3H): a single non-representative first step followed by convergence to a long-run $3\times$ cell-count increase per resolution level, here observed directly over ten resolution levels rather than assumed from the grid's name.

**To Complete / Verify**

The mechanism producing exactly a 7-fold, dedup-free first step and then convergence to 3-fold growth is not derived here from OpenEAGGR's underlying aperture-3 hexagon construction (`Src/Model/IGrid/IOffsetGrid/Aperture3HexagonGrid.cpp`); it is reported as an empirical regularity checked directly against the library's own enumeration API, not proved as a closed-form consequence of the construction. A closed-form growth law, analogous to DGGRID's $10 \cdot 7^n + 2$, remains an open item.

## 16.3 Area and compactness: results and comparison to the transcribed table

Resolutions 2 and 4 were chosen to match the labels used in Kmoch & Vasilyev (2022)'s transcribed OpenEAGGR table (Tables 11.1 and 11.2), following the same caveat already raised there for DGGRID: this chapter's resolution index is this chapter's own enumeration counter (Table 16.1), not confirmed to be the same index space as Kmoch et al.'s own OpenEAGGR configuration, only chosen as the two lowest levels giving a full-globe cell count in a plausible range for comparison.

Table 16.2: OpenEAGGR ISEA3H resolutions 2 and 4: this chapter's from-scratch computation (built from source, global, all 20 base faces, area/perimeter via `pyproj` WGS84 geodesics, 80 NaN-vertex cells excluded and counted at resolution 4) against Kmoch & Vasilyev (2022)'s transcribed values. Source: `code/eaggr_independent_reproduction.py`.

| res | source | $\sigma$(norm. area) | norm. area min | ipq mean | $\sigma$(ipq) | #cells |
|---|---|---|---|---|---|---|
| 2 | this chapter (independent) | 0.00773 | 0.9605 | 0.87686 | 0.05953 | 560 |
| 2 | Kmoch & Vasilyev (2022) (transcribed) | 0.00735 | 0.9606 | 0.87662 | 0.04243 | 862 |
| 4 | this chapter (independent) | 0.00334 | 0.9595 | 0.83183 | 0.11301 | 7,500 |
| 4 | Kmoch & Vasilyev (2022) (transcribed) | 0.00262 | 0.9636 | 0.87116 | 0.05993 | 45,589 |

**Key Result**

At resolution 2, this chapter's independent, from-scratch computation — sharing no code, no cell-generation library, and no area algorithm with Kmoch & Vasilyev's pipeline — agrees closely: $\sigma$(norm. area) 0.00773 vs. 0.00735 (5% relative difference), normalized-area minimum 0.9605 vs. 0.9606 (agreement to four significant figures), mean ipq 0.87686 vs. 0.87662 (0.03% relative difference). This is the closest resolution-2 agreement of any of the three from-scratch reproductions in this volume (H3, DGGRID, OpenEAGGR), and is the first time this volume's own spherical-excess and `pyproj` methods have been checked

against a *third*, independently authored geometric pipeline for the same nominal resolution and found to concur this closely on every reported statistic.

**To Complete / Verify**

Three gaps remain unexplained at resolution 4, in the same spirit as the unresolved gaps already disclosed for H3 and DGGRID. Chapters 17 and 18 test, respectively, the resolution-index reading offered for item (a) and the catastrophic-cancellation guess offered for item (c) below; neither survives its own test, and both chapters narrow rather than close these items. **(a) Cell count.** 7,500 valid cells here (of 7,580 enumerated) against Kmoch & Vasilyev's 45,589 — roughly a 6× gap, larger in relative terms than DGGRID's resolution-4 gap (24,012 vs. 23,608, under 2%) and in the opposite direction (this chapter under-counts relative to the transcribed figure, where for DGGRID it over-counted slightly). Combined with Table 16.1's observed ∼3×-per-level asymptotic growth, a 6× gap at a fixed nominal resolution label is consistent with Kmoch & Vasilyev's "resolution 4" being roughly one full ISEA3H refinement level finer than this chapter's resolution-4 enumeration counter — but this is only a plausible reading of the growth table, not a confirmed resolution-index mapping, and is reported as such rather than used to silently relabel either table's rows. **(b) The resolution-4 ipq mean.** 0.83183 here vs. 0.87116 transcribed is a larger discrepancy than the resolution-2 row shows, and larger than any other scheme's resolution-2-to-4 ipq drift documented in Chapter 11 or Chapter 15; whether this traces to the 80 excluded NaN-vertex cells having been disproportionately low-ipq outliers among the cells this chapter *did* manage to measure (plausible, since they sit at the icosahedron's singular vertices, structurally the most irregular cells in any ISEA tessellation, as already established for H3's and DGGRID's pentagons in Chapters 12 and 15) or to the resolution-index mismatch of item (a), or to some other cause, is not determined here. **(c) The method gap itself.** Unlike H3 and DGGRID, where the spherical-excess and `pyproj` methods agreed closely except at pole-adjacent cells, here the two methods disagree substantially even in aggregate: at resolution 2, spherical-excess gives $\sigma$(norm. area)$= 0.1235$ against `pyproj`'s 0.0077, a 16× gap (full figures in `data/eaggr_independent_reproduction_results.json`), far larger than the ∼1.8× gap seen for DGGRID at the same resolution. `pyproj`'s ellipsoidal geodesic engine is preferred throughout this chapter's headline table, consistent with the established practice of trusting the more numerically robust method rather than repairing the closed-form formula, but the size of this particular gap — an order of magnitude larger than for the other two schemes checked — is itself an unexplained finding, plausibly linked to catastrophic cancellation in the spherical-excess sum for OpenEAGGR's very small individual cell footprints, but not diagnosed further here.

## 16.4 What remains open

This closes Section 12.3's reproduction item for all five DGGS Kmoch et al. (2022) studied: H3 (Chapter 12), DGGRID (Chapter 15), and now OpenEAGGR (this chapter) have each received a from-scratch geometric reproduction; S2 and rHEALPix have instead received extensive real-tradeoff and operational-timing treatment (Chapters 95 to 101), a different but not lesser form of independent, from-scratch measurement. What is *not* yet done for OpenEAGGR, matching the gap already disclosed for DGGRID and S2 in Section 114.5: the real distortion-versus-storage measurement of Chapter 93 and the query-cost timing mea-

surement of Chapter 94, both of which would require this chapter's build of `libeaggr.so` to be driven through many resolution levels and repeated point/cell queries rather than the two-resolution geometric snapshot taken here. The three items flagged in the previous section's todobox — the resolution-index mapping, the resolution-4 ipq gap, and the anomalously large spherical-excess vs. `pyproj` disagreement — are also left open.

Chapter 17

# Testing the Resolution-Index Reading: Does an Integer Refinement Shift Explain the Gap?

## 17.1 Scope: turning a plausible reading into a checked claim

Section 16.3's todobox left item (a) unresolved as a *plausible reading*, not a checked claim: that Kmoch & Vasilyev's transcribed resolution-4 count of 45,589 cells might correspond to roughly one full ISEA3H refinement level finer than this chapter's own resolution-4 enumeration counter (7,580 deduplicated cells, Table 16.1), on the grounds that Table 16.1's observed $\sim 3\times$-per-level asymptotic growth would produce a comparable order-of-magnitude jump between adjacent levels. This chapter turns that reading into an explicit, checkable arithmetic claim against Table 16.1's own already-published counts, and reports whether it survives.

**Methodological Note**

**What was done, precisely.** No new cell enumeration, no new library call, and no new geometric computation of any kind. This chapter reuses exactly two already-published numbers: this volume's own deduplicated (and, for a secondary check, raw undeduplicated) ISEA3H cell counts by resolution from Table 16.1 (`data/eaggr_isea3h_growth_results.json`, unchanged), and Kmoch & Vasilyev's transcribed resolution-4 count of 45,589 cells (Table 16.2, unchanged). The only new computation is the ratio 45,589 divided by this chapter's resolution-4 count, compared against the single-level and double-level growth ratios already tabulated in Table 16.1. No cell counts are recomputed, re-measured, or adjusted; every number in Table 17.1 is either already published in this volume or a direct quotient of two such numbers.

## 17.2 The test

If Kmoch & Vasilyev's resolution-4 label corresponded exactly to this chapter's resolution 5 ("one level finer"), the ratio $45{,}589/7{,}580$ should match Table 16.1's own resolution $4 \to 5$ growth factor, 3.280. If it corresponded to resolution 6 ("two levels finer"), the ratio should match the compounded resolution $4 \to 6$ factor, $3.280 \times 3.163 = 10.372$.

**Key Result**

Neither counting convention supports the "one level finer" reading, and neither supports "two levels finer" either. Under the headline deduplicated counts, the target ratio (6.014) sits strictly between the single-level factor (3.280) and the double-level factor (10.372) — closer to the single-level factor in absolute terms but still 83% larger than it, and not reach-

Table 17.1: The resolution-index hypothesis tested arithmetically against already-published counts. "Target ratio" is 45,589 (Kmoch & Vasilyev's transcribed resolution-4 count) divided by this chapter's own resolution-4 count under each counting convention. "Nearest single-level factor" and "nearest double-level factor" are read directly from Table 16.1. Source: arithmetic only, on `data/eaggr_isea3h_growth_results.json` and Table 16.2.

| counting convention | this ch. res. 4 | target ratio | 1-level factor $(4 \to 5)$ | 2-level factor $(4 \to 6)$ |
|---|---|---|---|---|
| deduplicated (Table 16.1, headline) | 7,580 | 6.014 | 3.280 | 10.372 |
| raw, undeduplicated (secondary check) | 15,540 | 2.934 | 3.414 | 11.198 |

able by compounding any whole number of Table 16.1's own tabulated per-level factors from resolution 4 onward (which run $3.280, 3.163, 3.103, \ldots$, monotonically *decreasing*, so no later single step is closer to 6.014 than the first one). Under the raw undeduplicated counts — tried here only as a secondary check, since Section 16.2 already disclosed raw counts as the over-counted quantity, not a preferred alternative — the target ratio (2.934) falls *below* even the single-level factor (3.414), the opposite placement from the deduplicated case. The two counting conventions do not even agree on which side of the single-level factor the target ratio falls, which is itself evidence against treating either as a confirmed resolution-index correction rather than a coincidental order-of-magnitude match.

## 17.3 What this does and does not settle

This test does what Section 16.3's todobox explicitly declined to do: it converts the "roughly one full ISEA3H refinement level finer" reading from a qualitative impression into a specific numerical prediction, checks that prediction against already-published counts, and finds it does not hold under either counting convention tried. This narrows item (a) — an off-by-one resolution-index shift is no longer a live candidate explanation for the resolution-4 cell-count gap between this chapter and Kmoch & Vasilyev — but does not close it: no alternative explanation (a genuinely different ISEA3H parameterization on Kmoch & Vasilyev's side, a different base-cell or aperture-schedule choice, or some other configuration difference not visible from the transcribed summary statistics alone) is identified, tested, or ruled out here. The resolution-index mapping between this chapter's enumeration and Kmoch & Vasilyev's remains unresolved, now on a narrower and more specific footing than before.

Chapter 18

# Testing the Catastrophic-Cancellation Guess with Synthetic Hexagons

## 18.1 Scope: isolating one candidate mechanism

Section 16.3's todobox raised, but did not test, a candidate mechanism for item (c), the unusually large spherical-excess/`pyproj` disagreement observed for OpenEAGGR ($16\times$ at resolution 2, against $\sim 1.8\times$ for DGGRID at the same resolution): that it is "plausibly linked to catastrophic cancellation in the spherical-excess sum for OpenEAGGR's very small individual cell footprints". That guess makes a specific, testable prediction — that the disagreement should grow as cell size shrinks — which this chapter tests directly with synthetic hexagons, entirely independent of OpenEAGGR itself.

**Methodological Note**

**What was done, precisely.** No OpenEAGGR library call, no ISEA3H cell of any kind. Synthetic regular hexagons were generated directly on the WGS84 ellipsoid: for a given circumradius $r$ and a randomly chosen center (longitude uniform on $[-180, 180)$, latitude via $\arcsin(U(-1, 1))$ for area-uniform sampling on the sphere), six vertices were placed at bearings $0°, 60°, \ldots, 300°$ and geodesic distance $r$ from the center using `pyproj`'s `Geod.fwd` (WGS84 ellipsoid forward geodesic problem) — an independently specified construction sharing no code with any DGGS library. For each of nine circumradii, 400 such hexagons were generated (fixed seed 20260725, disclosed for reproducibility) and each was measured by the same two formulas already used throughout Chapter 16, copied verbatim from `code/eaggr_independent_reproduction.py` (not reimplemented differently): (a) the closed-form spherical-excess sum with haversine edges, (b) `pyproj`'s `Geod.polygon_area_perimeter`. For each formula and each radius, the normalized-area standard deviation $\sigma$ across the 400 hexagons was computed, exactly mirroring the statistic already reported in Table 16.2. Full code: `code/eaggr_method_gap_synthetic.py`; full output: `data/eaggr_method_gap_synthetic_results.json`.

**What this design does and does not reproduce.** Real ISEA3H cells are not perfectly regular hexagons: 12 of them (at the icosahedron's fixed vertices) are pentagons, and all of them carry whatever shape irregularity the ISEA3H projection and offset-grid construction impose, which is exactly what gives the real cells' $\sigma(\text{norm. area}) = 0.0077$ (`pyproj`, resolution 2, Table 16.2) its nonzero value in the first place. Synthetic hexagons built by construction to be regular have no such irregularity, so this design isolates *one* variable — circumradius, i.e. cell size — from the cancellation hypothesis specifically, at the cost of not reproducing real ISEA3H cell shape. This is disclosed as a limitation before the results are read, not after.

## 18.2 Results: the gap does not grow as cells shrink

Table 18.1: Synthetic regular-hexagon test, 400 hexagons per radius, $\sigma$(norm. area) by formula. Radii for the four "this chapter" rows are the circumradius of a disc of area equal to the mean cell area at that resolution (Earth surface area divided by cell count, Table 16.1), not an ISEA3H cell shape — a size proxy only. Source: `code/eaggr_method_gap_synthetic.py`.

| size label | radius (m) | $\sigma_{\text{spherical-excess}}$ | $\sigma_{\text{pyproj}}$ |
|---|---|---|---|
| res 2 proxy (this chapter, 560 cells) | 538,448 | 0.036104 | $1.17 \times 10^{-6}$ |
| res 4 proxy (this chapter, 7,580 cells) | 146,353 | 0.003969 | $8.61 \times 10^{-8}$ |
| res 6 proxy (this chapter, 78,620 cells) | 45,443 | 0.004007 | $8.49 \times 10^{-9}$ |
| res 9 proxy (this chapter, 2,267,020 cells) | 8,463 | 0.004061 | $3.4 \times 10^{-10}$ |
| synthetic 10 km | 10,000 | 0.003928 | $4.4 \times 10^{-10}$ |
| synthetic 1 km | 1,000 | 0.004234 | $4.6 \times 10^{-11}$ |
| synthetic 100 m | 100 | 0.004033 | $9.0 \times 10^{-10}$ |
| synthetic 10 m | 10 | 0.004070 | $8.75 \times 10^{-8}$ |
| synthetic 1 m | 1 | 0.003935 | $9.72 \times 10^{-6}$ |

**Key Result**

Two findings, both against the cancellation-for-small-footprints prediction. **First,** `pyproj`'s $\sigma$ stays at or below $\sim 10^{-6}$ across all nine radii spanning nearly six orders of magnitude (538 km down to 1 m), including at the radius matching real ISEA3H resolution 2 — four orders of magnitude below the real, measured resolution-2 figure of 0.0077 (Table 16.2). `pyproj` shows no cancellation-driven growth in error at small sizes here; its tiny residual $\sigma$ at every tested radius is consistent with ordinary floating-point noise on a perfectly regular construction, not a size-dependent numerical breakdown. **Second, and more directly against the guess:** the closed-form spherical-excess formula's $\sigma$ does *not* grow as radius shrinks. It is largest at the *largest* radius tested (0.036104 at 538 km, the res-2-scale proxy) and then sits in a narrow band (0.00393–0.00423, a $\pm 4\%$ range around 0.0040) essentially flat across every smaller radius tested, all the way down to 1 m — five further orders of magnitude of size with no monotonic trend in either direction. If catastrophic cancellation from small cell footprints were the driver of the observed spherical-excess/`pyproj` gap, $\sigma_{\text{spherical-excess}}$ should increase as radius shrinks; instead it is flat or, at the extreme, largest at the largest size.

**To Complete / Verify**

This test does not identify what *does* explain the closed-form formula's residual $\sigma \approx 0.004$ plateau, nor why it is roughly an order of magnitude larger at the res-2-scale radius specifically. A latitude-dependent or orientation-dependent bias in the spherical-excess formula's longitude-difference sum (rather than a cancellation effect) is a plausible reading of the flat-then-elevated pattern, but is not derived or confirmed here — it is offered only as the pattern actually observed, not as a mechanism verified against the formula's derivation. Nor does this test speak to whether real ISEA3H cell irregularity (pentagons, offset-grid distortion) interacts with the spherical-excess formula in some size-dependent way that this perfectly-regular synthetic construction cannot show; that remains untested.

## 18.3 What this does and does not settle

This chapter tests Section 16.3's catastrophic-cancellation guess directly, on its own stated terms — does the spherical-excess/`pyproj` gap grow as cell size shrinks — and finds no such growth across nearly six orders of magnitude of synthetic hexagon size, including at the exact size scale of the real resolution-2 measurement where the $16\times$ gap was originally observed. This narrows item (c): "very small cell footprint" is no longer a live explanation for the size of the observed gap, at least not through a raw floating-point cancellation mechanism of the kind this synthetic test isolates. It does not close item (c): the synthetic hexagons are, by construction, more regular than real ISEA3H cells, so a distinct mechanism tied to real cell irregularity rather than raw size — and the still-unexplained elevated $\sigma$ at the largest (res-2-scale) radius tested here — both remain open. The method-gap item, like the resolution-index item in Chapter 17, is left in a narrower but still unresolved state.

Chapter 19

# The Projections Behind the Grids: Snyder, HEALPix, and the Quadratic Cube

## 19.1 Scope: opening the black box

Every chapter of Part I so far has treated "ISEA", "rHEALPix", and "S2" as labels attached to a tessellation: a way of counting, indexing, and measuring cells. Chapters 11, 12, 15 and 16 reproduced cell geometry and compactness statistics for four of the five schemes surveyed by Kmoch et al. (2022) without ever asking what these labels stand for as *continuous-to-discrete maps*. Yet each name names, first and foremost, an explicit point-to-point projection from the sphere to a polyhedron (or the reverse): a map in the classical cartographic sense, of exactly the kind this volume's preface identifies as the object of study, only now applied face-by-face to a polyhedral solid rather than once to the whole sphere. This chapter recovers those maps from primary sources and asks, of each, the same question the continuous chapters of Monographs 1–4 asked of ordinary projections: what, geometrically, is preserved, and what is paid for it?

**Methodological Note**

This chapter contains no new computation and no data files. Every quantitative claim below is transcribed from a cited source and marked as such; nothing is derived from the code or data already in this volume. Where a source's numeric claim did not transfer cleanly to the specific configuration used elsewhere in this volume (e.g. Snyder's truncated-icosahedron bound versus the plain icosahedron used by ISEA-family DGGS), the mismatch is flagged in a To Complete/Verify box rather than silently reused.

## 19.2 The Snyder equal-area icosahedral projection (ISEA)

H3, DGGRID, and OpenEAGGR — three of the five schemes handled in Part I — all sit on the same base construction, usually compressed into the single acronym ISEA (Icosahedral Snyder Equal Area). Sahr et al. (2003) restate it in three steps: (1) inscribe a regular icosahedron in the sphere, in one of a small number of standard vertex orientations; (2) apply Snyder (1992)'s equal-area projection independently to each of the icosahedron's 20 equilateral-triangle faces, mapping the corresponding spherical triangle to the planar triangle face and back; (3) subdivide each planar triangle recursively (by edge trisection for the aperture-3 hexagonal case, or otherwise depending on the DGGS) and invert the same projection cell-by-cell to recover the tessellation on the sphere.

Snyder (1992)'s own construction is a modification of the ordinary Lambert azimuthal equal-area projection: each spherical triangle is treated not as a whole but split from its centroid into isosceles sub-triangles, each sub-triangle projected with a variant of the Lambert azimuthal map centred at the icosahedron face's own centre, and the results reassembled so that the triangle's edges match up exactly and the total area is preserved exactly. The projection is de-

liberately *not* conformal — it sacrifices local shape fidelity for exact area preservation, precisely the trade-off this volume's whole cost/distortion apparatus (Chapter 92) is built to quantify, except that here it is fixed by construction rather than tuned.

**Methodological Note**

Snyder (1992)'s own reported error bounds — angular deformation not exceeding 3.75° and scale variation under 3.3% — apply, by the abstract's own wording, to the *truncated* icosahedron (20 hexagon faces + 12 pentagon faces), which is a different base solid from the plain 20-triangle icosahedron used by H3, DGGRID's ISEA7H, and OpenEAGGR's ISEA3H. Reusing the truncated-icosahedron figures for the ISEA-family DGGS discussed elsewhere in this volume would misattribute a bound to the wrong polyhedron.

**To Complete / Verify**

The angular-deformation and scale-variation bounds specific to Snyder's projection applied to the *plain* regular icosahedron (as opposed to the truncated icosahedron) have not been located in the sources consulted for this chapter and are not asserted here. Locating them — most likely in Snyder (1992)'s full text rather than its abstract, or in a secondary source that treats the icosahedral case specifically — is left open for a future revision of this chapter.

This closes a gap silently carried through Chapters 11, 12, 15 and 16: those chapters measured the area and compactness *outcomes* of the ISEA construction empirically, cell by cell, without ever stating that exact equal-area behaviour is not an empirical regularity of those measurements but a guarantee built into Snyder's projection formula itself. The empirical near-uniformity in *compactness* (ipq) reported there, by contrast, is not guaranteed by the projection and remains a genuinely empirical finding.

## 19.3 The rHEALPix projection: cylinder, cap, and a rearranged cube

rHEALPix takes a different base solid — the cube, not the icosahedron — and a different projection lineage, rooted in astronomy rather than cartography proper. Górski et al. (2005) defined HEALPix (Hierarchical Equal Area isoLatitude Pixelisation) to tile the sphere for storing and analysing cosmic microwave background data; Gibb et al. (2016) rearranged its geometry, without altering the underlying projection mathematics, into a form whose planar layout is a genuine cube net rather than HEALPix's original four-armed diamond shape, in order to meet the OGC's DGGS requirements for a regular polyhedral base solid with square, array-storable faces.

The projection itself, as derived rigorously by Calabretta & Roukema (2007), is a hybrid of two classical pieces, joined at a latitude fixed not by convention but by an area-equality argument:

- **Equatorial belt** ($|\theta| \leq \theta_\times$): an ordinary Calabretta & Roukema (2007)-attributed Lambert (1772) cylindrical equal-area projection — meridians and parallels of the sphere map to a regular rectangular grid, exactly as in the classical cylindrical equal-area case treated in Monographs 1–4's continuous framework.
- **Polar caps** ($|\theta| > \theta_\times$): an interrupted projection in the lineage of Collignon (1865), in which meridians converge to a point at each pole and parallels are unequally spaced, again constructed to be exactly area-preserving.

The transition colatitude $\theta_\times$ is not a free parameter but is fixed by requiring the two pieces to meet with matching area density. Calabretta & Roukema (2007) derive it by equating the area of a spherical cap, $A = 2\pi(1 - \sin\theta)$, with the fixed $4\pi/6$ share of the sphere's total area occupied by each of the six polar half-facets in HEALPix's base $N_{\text{side}} = 1$ tiling, independent of resolution:

$$\sin\theta_\times = \frac{2}{3}, \qquad \theta_\times = \sin^{-1}\left(\frac{2}{3}\right) \approx 41.8103^\circ.$$

Because rHEALPix inherits this projection unmodified from HEALPix (only the planar arrangement of facets is rearranged into a cube net), the same transition colatitude governs where every rHEALPix cell crosses from cylindrical to Collignon-type behaviour, regardless of the DGGS's resolution parameter.

**Key Result**

Unlike ISEA, whose exact equal-area guarantee comes from a single integrated projection formula applied per triangular face, rHEALPix's exact equal-area guarantee is assembled from two *different* classical projections (cylindrical equal-area and a Collignon-type polar projection), stitched at a latitude, $\theta_\times = \sin^{-1}(2/3)$, chosen precisely so that the area density matches across the seam. Both schemes are exactly equal-area; they reach it by structurally different routes, one continuous and one piecewise. This is a genuinely cartographic distinction, prior to and independent of any implementation or resolution parameter.

## 19.4 The S2 quadratic transform: uniformity bought, not guaranteed

S2 breaks the pattern of the first two sections in a way this volume has not yet stated explicitly: it is *not* an equal-area scheme by construction. Its base solid is again a cube, but the six faces are projected onto the sphere by first mapping a face to cube-space coordinates $(u, v) \in [-1, 1]^2$ via gnomonic (central) projection, then correcting for the severe area non-uniformity that a plain gnomonic cube projection produces. Google (2024)'s own developer documentation states the problem plainly: a naive linear parametrisation leaves the largest cells (at face centres) several times larger in spherical area than the smallest (at face corners).

The correction is a deliberately *inexact* one, chosen for speed rather than for an exactness guarantee. The same documentation describes, and Google's own public presentation of the library quantifies, three candidate nonlinear transforms $u = f(s)$ mapping cell-space $(s, t) \in [0, 1]^2$ to cube-space $(u, v)$ before the final gnomonic-style projection onto the sphere:

| Transform | Max area ratio (largest/smallest) | Relative cost |
|---|---|---|
| Linear | 5.2 | fastest |
| Quadratic (S2 default) | 2.08 | cheap polynomial |
| Tangent (arctan-based) | 1.41 | $\sim 3\times$ slower |

**Methodological Note**

These three figures are transcribed directly from Google (2024)'s public technical presentation of the library and cross-checked against the `s2.h` source comments of the archived reference implementation, both consulted directly for this chapter. They are not independently recomputed here. The explicit polynomial coefficients of the quadratic transform itself are not quoted, since no primary formula was located and verified in this pass; as-

serting one from memory would risk misstating the exact function. This is left as an open item rather than filled with an unverified formula.

The comparison table makes explicit what the tangent transform's own name concedes: it is the one closest to a true equal-area correction (ratio 1.41, close to the geometric ideal of 1), obtained precisely by using the trigonometric function that the quadratic transform is built to *approximate* without paying for it. S2's own documentation states that the resulting per-cell area is guaranteed only to within an accuracy band (approximate area within 3% for all cell sizes, exact area computable but expensive), never to be exactly uniform or exactly equal-area at any level, in contrast to both ISEA and rHEALPix.

**Key Result**

S2 is the one scheme, of the three examined in this chapter, whose projection is explicitly *not* equal-area: uniformity of cell size is bought approximately, by a cheap polynomial standing in for a more exact but costlier trigonometric correction, and the resulting inexactness is disclosed by the library's own documentation as a quantified error band rather than eliminated. Compared against Sections 19.2 and 19.3, this is not a difference of *degree* in implementation quality but a difference in the basic cartographic contract each scheme keeps with the user of its grid.

## 19.5 Synthesis: three answers to one cartographic question

Stripped of implementation detail, Sections 19.2 to 19.4 answer one shared question in three structurally distinct ways: given a polyhedral solid whose faces are to be recursively subdivided, what continuous sphere-to-face map should be used to *define* the subdivision?

| Scheme | Base solid | Projection lineage | Equal-area? |
|---|---|---|---|
| H3 / DGGRID / OpenEAGGR | Icosahedron (20 tri.) | Snyder (1992), modified Lambert | Exact, by construction |
| rHEALPix | Cube (rearranged HEALPix) | Lambert cyl. + Collignon-type | Exact, piecewise |
| S2 | Cube (gnomonic) | Gnomonic + quadratic correction | Approximate only |

This table could not have been built from anything already in this volume: Chapters 11, 12, 15 and 16 measured area and compactness *outcomes* empirically, cell by cell, without stating which of those outcomes were guaranteed in advance by the projection formula and which were genuinely open empirical questions. The answer, now available: for ISEA and rHEALPix, area equality is not a finding but an axiom of the projection; for S2, it is not even an axiom, only a controlled approximation, which is exactly why an S2-based DGGS requires a separate empirical area accounting in a way that an ISEA-based or rHEALPix-based one does not by the same necessity. This is precisely the distinction the preface's opening question — how much does it cost to represent something continuous discretely? — was asking for, now answered at the level of the projection formula itself rather than at the level of a measured cell statistic.

**To Complete / Verify**

Two natural continuations of this chapter are deliberately left for separate treatment rather than folded in here: (i) the question of how the base polyhedron's *orientation* relative to the Earth's actual geography (as opposed to its abstract construction) affects total distortion over the continents actually being mapped, taken up next; and (ii) any timing or implementation-level behaviour of these projections, which belongs with Part III and not with this chapter's purely cartographic scope.

Chapter 20

# Why a Cube: S2 Against a Longer Cube-Sphere Tradition

## 20.1 Scope: a question the synthesis table left open

Section 19.5's synthesis table closed Chapter 19 by setting three schemes side by side: H3/DGGRID/OpenEAGGR on an icosahedron via Snyder's modified Lambert azimuthal projection, rHEALPix on a cube via a Lambert-cylindrical/ Collignon hybrid, and S2 on a cube via a gnomonic projection corrected by a cheap polynomial. That table recorded *what* each scheme does; it did not ask why S2's designers reached for a cube and a gnomonic-style projection in the first place, rather than adopting the icosahedral, exactly-equal-area construction Sahr, Snyder, and the ISEA-family DGGS had already settled on by the 1990s. This chapter answers that question, and the answer runs through a project with nothing to do with either digital maps or Google: a 1975 U.S. Navy weather-data study.

**Methodological Note**

This chapter introduces four new sources to Section 114.6: PROJ contributors (2026) and Veach (2017), both fetched and read directly in this pass; and Chan & O'Neill (1975) and O'Neill & Laubscher (1976), cited only via PROJ contributors (2026)'s own bibliographic description of them and not fetched directly (both are 1970s technical reports, not journal articles, and this pass does not attempt to retrieve the archive.org copies PROJ contributors (2026) links to). No new computation is performed; the S2-specific figures reused below (the three-way transform comparison table) were already established, from Google (2024), in Chapter 19.

## 20.2 A cube-sphere lineage that predates S2 by three decades

PROJ contributors (2026) documents a specific projection family, the *Quadrilateralized Spherical Cube* (QSC), built for exactly the purpose Section 19.4 described S2 as solving: mapping the sphere onto the six faces of a cube so that data can be organised in a quadtree per face. PROJ contributors (2026) traces the QSC projection to O'Neill & Laubscher (1976), building on earlier work by Chan & O'Neill (1975) — a feasibility study, per its own title, of a "quadrilateralized spherical cube earth data base," commissioned not for cartography or computing as such but for the U.S. Navy's Environmental Prediction Research Facility, evidently to store gridded meteorological data. PROJ contributors (2026) notes that this original 1975 variant later became known as the "COBE QSC," because NASA's Cosmic Background Explorer satellite mission adopted the same cube-face tessellation in the late 1980s to store its all-sky microwave background maps — a second, independent reuse of the same construction, this time for cosmology rather than meteorology or terrestrial cartography, more than a decade before S2 existed.

**Key Result**

The cube, as a base solid for a quadtree-indexed discrete global grid, is not an idea original to S2 or to Google: PROJ contributors (2026)'s account puts a fully worked cube-sphere quadtree scheme in active use for gridded geophysical data by 1975, and in active use for satellite cosmology data by the late 1980s, both well before S2's public open-source release in December 2017 — and, per that release announcement's own account, even before an early, less complete version of the S2 code that Veach (2017) reports was already circulating internally in 2011 — and before H3's icosahedral aperture-7 hexagonal scheme was developed. When Section 19.5 set S2's cube against H3's icosahedron, it was implicitly setting one multi-decade design tradition against another, not a novel Google-specific choice against an established academic norm.

## 20.3 What the earlier lineage solved that S2 does not

The detail that makes this lineage more than a historical footnote is that O'Neill & Laubscher (1976)'s 1976 refinement of the original 1975 scheme was already, per PROJ contributors (2026), an *equal-area* projection: the QSC projection proper (as opposed to the cruder "COBE QSC" variant of the 1975 report) "is an equal-area projection, and at the same time introduces only limited angular distortions," treating all six cube faces identically rather than singling out any face for special treatment near a pole or an edge (PROJ contributors, 2026). That is, an exact solution to the specific problem Section 19.4 showed S2 solving only approximately — uniform cell area across a cube-face tessellation — was already published, worked out to the level of an implementable formula, more than forty years before this volume's own account of S2's quadratic transform and its documented 3% area-accuracy band.

PROJ contributors (2026) states plainly that a designer choosing how to project a sphere onto a cube today has more than one option: "other alternatives can be used, notably Gnomonic or HEALPix" alongside QSC itself. S2, on the evidence assembled in Chapter 19 and here, took the gnomonic branch of that choice and then corrected it only partially, with the cheap quadratic transform rather than the more expensive, more nearly equal-area tangent transform the same chapter's comparison table already recorded. Read against O'Neill & Laubscher (1976)'s earlier, exact solution, S2's choice is not evidence that a good cube-face equal-area construction was unknown or unavailable; it is evidence that S2's designers, per Google (2024)'s own documented trade-off, valued the polynomial's speed over the exactness a forty-year-old formula could already have supplied.

## 20.4 Why a cube at all: the quadtree constraint

This chapter's title asks "why a cube," and the material above answers only half of it: it establishes that cube-face tessellation for quadtree-organised spherical data is an old and repeatedly reused idea, not why that idea, rather than an icosahedral one, is the right fit for a spatial-indexing system specifically. PROJ contributors (2026) states the reason directly, in the same sentence that motivates QSC itself: cube-face data "can be organised in quadtree structures for each cube side." A quadtree requires each cell to subdivide into exactly four children of the same shape as their parent; a square face subdivides into four smaller squares this way without remainder, recursively, to any depth. A triangular icosahedron face does not: the ISEA family's own recursive subdivision, reused throughout Part I of this volume, proceeds by a different rule (edge trisection into nine children for the aperture-3 case that underlies H3's

hexagons, per Chapter 25's own account), incompatible with a four-way quadtree and with the Hilbert-curve linearisation Google (2024) builds on top of S2's quadtree specifically for its 64-bit cell-ID indexing scheme.

**To Complete / Verify**

This chapter asserts, without independently verifying it against a primary source on H3's or DGGRID's own indexing internals, that the icosahedral ISEA family's aperture-3/aperture-7 subdivision rule is structurally incompatible with a four-child quadtree in the same sense S2's cube-face subdivision is compatible with one. This is stated on the strength of Chapter 25's existing account of the aperture hierarchy and ordinary quadtree/quadrilateral-subdivision facts, not on a source that discusses H3's or DGGRID's cell-indexing scheme directly by name; a future revision of this chapter should locate and cite such a source before treating the claim as settled.

## 20.5 Closing: two lineages, two design priorities

Section 19.5's table can now be read with an extra row of context behind it. H3, DGGRID, and OpenEAGGR inherit Snyder's 1992 icosahedral equal-area construction because their designers' priority, per Chapter 19, was an *exact* area guarantee purchased at the cost of a base solid (the icosahedron) that does not admit a simple quadtree. S2 inherits a cube because its designers' priority was quadtree- and Hilbert-curve-based spatial indexing at global scale, a structural requirement the cube satisfies and the icosahedron does not, and only within that already-settled choice of base solid did the further choice arise between an exact area-preserving cube projection (available since 1976) and a cheaper, approximately area-preserving one (the choice S2 actually made). "Why gnomonic" turns out to be the wrong question, or at least a question asked one level too low: the gnomonic-plus-correction family is simply the cube-projection default absent a specific reason to pay for QSC's exactness, and the decision that mattered more was made earlier, when quadtree indexability, not distortion minimisation as such, selected the cube over the icosahedron in the first place.

Chapter 21

# Orienting the Polyhedron: A Second, Independent Cartographic Choice

## 21.1 Scope: a design axis this volume has not yet touched

Chapter 19 isolated the *projection* that maps a polyhedron's faces to the sphere as one design choice among several, fixed independently of any resolution parameter or implementation detail. But Sahr et al. (2003) and Barnes (2019) both identify a second, logically independent choice, made before any projection formula is even applied: given a base polyhedron, *where* is it placed relative to the Earth's actual geography? A regular icosahedron has no privileged position on the sphere — any rotation of it is equally regular — so this placement is a free parameter of the construction, fixed once by convention for a given DGGS and then inherited by every resolution built on top of it. This chapter treats that placement as what it is: a self-contained cartographic optimization problem, with a documented history and a rigorous modern solution, entirely independent of the tessellation counting and timing questions that occupy Part III.

**Methodological Note**

As in Chapter 19, this chapter performs no new computation. Every numeric value below, including the orientation table in Section 21.3, is transcribed directly from Barnes (2019)'s published results and attributed as such.

## 21.2 Before optimization: Fuller's eye and an inconsistent legacy

The oldest documented icosahedral orientation is not the product of optimization at all. Buckminster Fuller's Dymaxion map, unfolding the triangular faces of an icosahedron, was oriented so that none of its twelve vertices fell on a landmass — a property Sahr et al. (2003)'s tradition treats as Fuller's map's own claim to fame, and one which, according to Barnes (2019), remained the *only* known icosahedral orientation with that property until formal optimization methods were applied to the problem seven decades later. Fuller's orientation was found by inspection, not computation, and it shows: Barnes (2019) reports that published tables of Fuller's own vertex coordinates disagree with each other, and some are internally inconsistent, a state of affairs traced by Barnes (2019) to a 2015 survey of the available sources. Producing a single self-consistent set of coordinates for Fuller's orientation, verified against a stated optimization criterion rather than read off a hand-drawn globe, was itself one of Barnes (2019)'s results, not a solved problem before that paper.

This is worth pausing on for what it says about the maturity of an apparently simple design choice: seventy years after the Dymaxion map's introduction, no algorithmically verifiable, internally consistent statement of its own defining orientation existed in the literature. The two other historically used icosahedral orientations named by Sahr et al. (2003)'s tradition — one,

due to White et al. (1992), built for a truncated icosahedron rather than the plain 20-triangle solid used by ISEA-family DGGS; the other, the orientation used by Sahr et al. (2003)'s own software lineage, placing exactly one vertex on a continent rather than zero — were likewise convention-driven choices, not outputs of a stated optimization criterion.

## 21.3 Orientation as a three-parameter optimization problem

Barnes (2019) formalizes the placement of a polyhedron with exactly three real parameters: the latitude $\lambda$ and longitude $\varphi$ of a chosen "primary" vertex, and a rotation $\theta \in [0°, 360°)$ of the remaining vertices about the axis running from the sphere's centre through that primary vertex. Any orientation of the icosahedron — Fuller's, Sahr's, or any other — is a single point $(\lambda, \varphi, \theta)$ in this three-dimensional space, and the question of what makes an orientation "good" becomes an ordinary, if nonconvex, optimization problem over that space once a criterion is fixed.

Barnes (2019) solves it for several criteria relevant to any DGGS built on a polyhedron: (a) maximizing or minimizing the *average* distance of all vertices from the nearest coastline, which pushes vertices — and the singular cells that sit on them — toward open ocean or toward land, respectively; (b) maximizing or minimizing the kilometres of polyhedron *edges* that overlap landmasses, which controls how much of a continent's interior is cut by a resolution-defining boundary; and (c) maximizing a single vertex's distance from any coastline, which yields the four terrestrial and oceanic Poles of Inaccessibility as a side effect of the same method. All three are solved by simulated-annealing hillclimbing over real coastline data (OpenStreetMap and GSHHG), not by inspection.

| Objective | Vertices on land | Value (km) | $\lambda$ | $\varphi$ | $\theta$ |
|---|---|---|---|---|---|
| Max. avg. dist. from coast | 0 | 717.62 | 25.4981 | −139.4880 | 58.2846 |
| Max. avg. dist. from coast | 8 | 407.80 | 21.8598 | −14.1591 | 47.9955 |
| Min. avg. dist. from coast | 0 | 223.73 | 24.9672 | 67.4845 | 57.7353 |
| Min. avg. dist. from coast | 8 | 44.18 | 34.5965 | −85.5585 | 34.8110 |
| Max. edge overlap w/ land | 0 | 3147.00 | 55.0965 | 140.7330 | 15.5157 |
| Max. edge overlap w/ land | 8 | 3576.00 | 79.1494 | −54.0237 | 49.0449 |
| Min. edge overlap w/ land | 0 | 2456.00 | 18.9602 | −61.1569 | 21.7798 |
| Min. edge overlap w/ land | 8 | 3007.00 | 37.3477 | −85.0918 | 33.5892 |

**Methodological Note**

This table transcribes only the eight regular-icosahedron rows of Barnes (2019)'s Table 2, restricted to the criteria most directly relevant to a DGGS designer (average coastal distance and edge overlap). The paper's parallel tables for the cuboctahedron, dodecahedron, octahedron, and tetrahedron, and its separate table of the four Poles of Inaccessibility with sub-kilometre precision, are not reproduced here as out of scope for an icosahedron-focused volume. "Vertices on land" counts how many of the twenty icosahedral vertices fall inside a landmass polygon under that orientation; 0 reproduces Fuller's defining property by construction, verified rather than assumed, while 8 characterizes an orientation deliberately chosen for the opposite purpose.

**Key Result**

Barnes (2019)'s own comparison closes the loop opened in Section 21.2: of the several zero-vertices-on-land orientations the optimization finds, the one that *also* minimizes edge overlap with landmasses is the one that most closely resembles Fuller's historical orientation. Fuller's map was, in this precise sense, close to optimal along the one axis it was informally designed for — without ever having been verified as such, and without having been optimized along the orthogonal axis (edge overlap) that a formal method reveals as a separate, independently tunable quantity.

## 21.4 Why this matters for a tessellation, not just a flat map

Fuller's map is a flat, interrupted world map; nothing in Sections 21.2 and 21.3 depends on that. The same orientation parameters $(\lambda, \varphi, \theta)$ apply identically to the spherical polyhedron underlying any ISEA-family DGGS before its faces are ever subdivided into cells. This is where Section 21.3 connects directly to results already established empirically, rather than theoretically, elsewhere in this volume.

Kimerling et al. (1999), as restated by Barnes (2019), establish that a hexagonally-tiled icosahedron has *exactly* twelve pentagonal cells at every resolution, one centred on each of the icosahedron's twelve vertices, each pentagon exactly $5/6$ the area of a surrounding hexagon. Chapters 12, 15 and 16 of this volume already found this structural fact empirically and independently, three separate times, for H3, DGGRID, and OpenEAGGR: each reproduction turned up exactly twelve unaccounted-for or structurally distinct cells, traced in each case to the icosahedron's fixed vertex population. What none of those three chapters asked — because it was, at that point, outside their scope — is *where on the Earth's surface* those twelve singular cells, and the areal or compactness anomalies that Chapter 11's own tables already recorded around them, actually sit. Section 21.3's answer is that this is not a fixed fact about ISEA-family DGGS at all, but a free consequence of a choice made once, upstream of any resolution: an orientation from the maximize-average-distance row of the table above concentrates the twelve pentagons and their neighbouring distortion over open ocean, while an orientation from the minimize-average-distance row concentrates the same twelve singularities directly over populated landmasses.

**To Complete / Verify**

Whether H3, DGGRID's default ISEA7H configuration, and OpenEAGGR's default ISEA3H configuration — the three schemes independently reproduced in Chapters 12, 15 and 16 — actually use an orientation matching any row of Barnes (2019)'s table, Fuller's orientation, or Sahr et al. (2003)'s one-vertex-on-land convention has not been checked in this volume. Chapter 23 closes this for H3: its twelve base-cell vertices match none of Barnes (2019)'s four zero-land rows but match Fuller's own orientation to within metres. DGGRID's and OpenEAGGR's default orientations remain unchecked, both requiring compiled binaries not available in this environment, and are left open rather than assumed.

## 21.5 Synthesis: two independent axes, not one

Chapters 19 and 21 together make a point neither could make alone: a DGGS's cartographic behaviour is governed by (at least) two design choices that are logically independent of each

other and of any implementation detail. The projection formula (Chapter 19) fixes what happens *within* a face, independent of where that face sits on the globe; the orientation (this chapter) fixes *where* the faces, and their vertex singularities, sit relative to the actual geography being measured, independent of how a face's interior is projected. A change to either axis alone changes the resulting DGGS without touching the other. Neither axis is a matter of software engineering or computational efficiency; both are settled entirely before any cell is counted, any resolution is chosen, or any timing benchmark of the kind treated in Part III could even be run.

Chapter 22

# The Polyhedral Tradition Before and After Fuller

## 22.1 Scope: Fuller's map was not the first, and the icosahedron was not the only choice

Chapter 21 examined Buckminster Fuller's Dymaxion map in isolation, as a single, historically fixed orientation of the icosahedron whose own published coordinates disagree across sources (Section 21.2). That treatment took the choice of polyhedron itself — the icosahedron, the same twenty-triangle solid underlying the ISEA-family DGGS this volume has followed since Chapter 19 — as given. This chapter steps back one level further, to ask where that choice sat within a longer cartographic tradition of unfolding a globe onto a polyhedron to reduce interruption and distortion, a tradition that neither began with Fuller in 1943 nor settled definitively on the icosahedron. A peer-reviewed survey of that tradition puts it plainly: polyhedra have served as cartographic image surfaces for over two hundred years, and the Platonic solids are only one family within a considerably larger design space that includes Archimedean and other semi-regular solids (Pędzich, 2017).

**Methodological Note**

This chapter introduces two new sources to Section 114.6 (Pędzich, 2017; Cahill, 1909) and one tertiary source flagged explicitly as such (Wikipedia contributors, 2026), consistent with this volume's existing practice of using a flagged tertiary source only for well-attested, non-numerically-sensitive historical facts (compare Wagner, 2022 in Chapter 24). No new computation is performed; every figure below (patent year, octant edge length, publication years) is attributed to a specific source at point of use.

## 22.2 Cahill's octahedron, 1909–1913

Three decades before Fuller, the American cartographer and architect Bernard J. S. Cahill published a "Butterfly Map" built on the regular octahedron rather than the icosahedron: eight symmetrical triangular lobes, four above and four below the equator, arranged so that every continent could be shown uninterrupted on the unfolded two-dimensional map (Pędzich, 2017; Cahill, 1909). Cahill's own 1909 account of the design, published in *The Scottish Geographical Magazine*, laid out the octahedral construction in full technical detail across twenty pages (Cahill, 1909); the phrase he later used to summarize that same construction, "decentralised octahedral system of projection," is instead securely attested from a 1932 exchange discussed in Section 22.3 (Cahill, 1932), and is deferred to there rather than quoted from the 1909 article, which was not fetched in this pass. The design was granted U.S. Patent 1,054,276 in 1913 (Wikipedia contributors, 2026). Cahill also built a physical demonstration of the same idea already familiar to this volume in the abstract, as Chapters 1 and 19's recurring continuous-

versus-discrete theme: a rubber-ball globe that could be flattened under a pane of glass into the butterfly arrangement and would return to its spherical shape when released (Wikipedia contributors, 2026) — a physical, elastic model of exactly the projection-then-unfolding operation this volume's own DGGS chapters perform mathematically.

**To Complete / Verify**

Cahill's original 1909 article is available in full via a dedicated resource site maintained by Gene Keyes, but was not fetched or quoted from in this pass; only its existence, title, venue, and length (pp. 449–469), attested consistently across Pędzich (2017)'s own reference list and the resource site's framing, are used here. The "decentralised octahedral system" phrase used elsewhere in this chapter is sourced separately, to the 1932 *Nature* exchange (Section 22.3), not to this 1909 article.

## 22.3 A 1932 exchange, and the octahedron's persistence

The Butterfly Map remained a live topic of technical debate for decades after its introduction. A short exchange in *Nature* in 1932 records Cahill himself responding to what he considered a misreading, in an earlier issue of the same journal, of a combined diagram in which he had overlaid three variants of his projection into a single figure; Cahill's reply refers to "tabulated computations and diagrams of his decentralised octahedral system of projection begun thirty years ago," explicitly dating his own work's origin to the early 1900s and confirming the octahedral (not icosahedral) basis of the design he was still defending in print more than twenty years after the original patent (Cahill, 1932). This is worth recording alongside Section 21.2's account of Fuller's own inconsistent published coordinates: both polyhedral map traditions generated genuine, published, contemporaneous technical disagreement about their own construction, a pattern this volume's insistence on independent reproduction (Chapters 12, 15 and 16) is implicitly a response to, whether or not any DGGS designer discussed elsewhere in this volume was aware of either historical episode.

## 22.4 After Cahill: refining the same octahedron, twice

Cahill's octahedral idea was not abandoned after his death in 1944; it was refined twice more, each time by a different author, without changing the underlying polyhedron.

**Cahill–Keyes (1975).** Gene Keyes proposed a refinement of Cahill's 1909 projection with several explicit design goals stated in the projection's own documentation: symmetry among the eight octants, scalability so the map remains usable at high resolution, uniform geocells, and — the property most directly comparable to this volume's own concerns — metric-based joining edges, with each octant's three principal edges fixed at a stated length of 10,043 km so that adjacent octants meet consistently (Wikipedia contributors, 2026). This is the same edge-matching discipline Chapters 11 and 21 already require of a DGGS's own cell boundaries, applied here to eight large octant faces instead of millions of small DGGS cells, decades before any of the DGGS constructions surveyed elsewhere in this volume were built.

**Waterman butterfly (1996).** Steve Waterman's variant replaced Cahill's plain octahedron with a *truncated* octahedron, again described as evoking "the butterfly map principle first developed by Bernard J. S. Cahill ... in 1909" (Wikipedia contributors, 2026). The move from a Platonic solid (the plain octahedron) to an Archimedean one (the truncated octahedron) for a butterfly-style map by 1996 anticipates, in ordinary cartography, the same move Chapter 19

already documented for DGGS design: White et al. (1992)'s truncated icosahedron, used for the EMAP sampling grid discussed at length in Chapter 72, is the same kind of refinement — an Archimedean truncation of a Platonic solid, chosen for a better approximation to the sphere — applied to the icosahedron instead of the octahedron, and for a sampling design rather than a display map.

**Key Result**

Three cartographic lineages surveyed across this volume — Cahill's octahedron and its Keyes and Waterman descendants (this chapter), Fuller's plain icosahedron (Chapter 21), and the truncated icosahedron and other ISEA-family solids underlying modern DGGS (Chapters 11 and 19) — all pursue the identical goal by the identical means: choose a polyhedron whose faces approximate the sphere, unfold or index it with a consistent rule for matching adjacent faces, and accept a fixed, enumerable set of interruption or singularity points (Cahill's eight octant vertices, Fuller's icosahedron vertices, the twelve pentagonal DGGS cells Chapter 21 already documents) in exchange for uninterrupted, low-distortion coverage everywhere else. The choice of *which* polyhedron — octahedron, icosahedron, their truncated Archimedean refinements — has never been unique, and this volume's own preference for the icosahedron (Chapter 19) is a documented, quantified instance of a choice cartographers were already making, and disputing, a century earlier with fewer faces and no computational verification available to them.

## 22.5 Why the icosahedron eventually displaced the octahedron for this volume's purposes

Nothing in Sections 22.2 and 22.4 claims that Cahill's octahedron was geometrically inferior to Fuller's icosahedron in any sense this volume has already quantified; no source consulted for this chapter reports an isoperimetric quotient, an area or shape distortion figure, or any other measure of Table 11.2's kind for Cahill's, Keyes's, or Waterman's specific octant faces. What can be said, using only what this volume has already established, is the structural reason a modern DGGS designer would reach for twenty faces rather than eight in the first place: Chapter 19 already establishes that the ISEA family's twenty equilateral-triangle icosahedron faces are each smaller, and individually cover a smaller angular extent of the sphere, than any one of an octahedron's eight faces would; a smaller face, projected by the same class of equal-area method, departs less from its own tangent plane and so accumulates less internal distortion before reaching its own edge. This is precisely the logic Chapter 19's own citation record (Snyder, 1992, discussed there) already gives for choosing the icosahedron over coarser Platonic solids for the ISEA construction; this chapter's contribution is only to show that the coarser alternative Snyder's generation of designers moved past was not a straw man but Cahill's specific, patented, and repeatedly refined 1909 design, still being built and sold as recently as Waterman's 1996 variant.

**To Complete / Verify**

No source consulted in this pass reports a directly computed distortion or compactness figure for Cahill's, Keyes's, or Waterman's octahedral butterfly maps in units comparable to Table 11.2's isoperimetric quotients. The qualitative structural argument in this section (smaller face, proportionally less accumulated distortion) is this volume's own inference from already-established geometric facts about face count and angular extent, not a figure transcribed from either Cahill's, Keyes's, or Waterman's own publications, and should not

be read as a substitute for an actual computed comparison, which this chapter does not attempt.

## 22.6 Synthesis: a two-hundred-year design space, of which this volume documents one corner

Pędzich (2017) frame the entire polyhedral-map tradition as more than two centuries old, running from early nineteenth-century proposals through Cahill's octahedron, Fuller's icosahedron, and Waterman's truncated octahedron, to — in cartography's most recent turn — the hierarchical, computationally indexed DGGS constructions this volume has spent Part I reproducing and comparing. Read against that longer history, Chapters 1, 19 and 21's treatment of H3, S2, rHEALPix, DGGRID, and OpenEAGGR is not a break from earlier cartography but a continuation of it under a much stricter standard of verification: where Cahill defended his octant geometry in a magazine exchange (Section 22.3) and Fuller's own vertex coordinates still disagree across published tables (Section 21.2), this volume's own reproduction chapters recompute cell counts, areas, and compactness figures independently from primary code and data (Kmoch & Vasilyev, 2022, throughout Chapter 11 and the chapters that follow it) precisely so that the same kind of unresolved disagreement does not persist for the constructions examined here. The polyhedron changed twice more after Cahill — Fuller's icosahedron, then the ISEA family's own refinements of it — but the underlying cartographic problem, and the polyhedral answer to it, is the same one Cahill was already patenting in 1913.

Chapter 23

# Confirming H3's Orientation: The Fuller Correspondence, Verified

## 23.1 Scope: closing one branch of Chapter 21's open question

Chapter 21's todobox left open whether H3, DGGRID's default ISEA7H configuration, and OpenEAGGR's default ISEA3H configuration actually use an orientation matching any row of Barnes (2019)'s table, Fuller's orientation, or Sahr et al. (2003)'s one-vertex-on-land convention. This chapter answers that question for H3, the one of the three schemes whose Python bindings install without a compiled toolchain (Chapter 12's own `h3` library, not previously used elsewhere in this volume). DGGRID and OpenEAGGR both require binaries this environment does not have available and remain open, narrowing rather than closing Chapter 21's todobox.

**Methodological Note**

Unlike Chapters 21 and 22, this chapter performs new computation. It reproduces Barnes (2019)'s own three-parameter placement construction (Sec. 2.2 of that paper: a nominal icosahedron vertex table, a spin by $\theta$ about the polar axis, then a Rodrigues rotation carrying the polar axis onto the target primary vertex) independently from the paper's description, and verifies the reconstruction against the one case that must reproduce the nominal table exactly ($\lambda = 90°, \varphi = 0°, \theta = 0°$, recovering all twelve nominal vertices to within $10^{-6}$ degrees, floating-point noise). Land/ocean status is checked using `global_land_mask`, a bundled 1°-resolution raster — an independent data source from both Barnes (2019)'s OpenStreetMap/GSHHG coastlines and any dataset used elsewhere in this volume, disclosed as such because its coarser resolution is adequate for a binary on-land/in-ocean check but would not be adequate for the kilometre-precision distance optimization that chapter performs. Full code: `code/h3_orientation_check.py`; full output: `data/h3_orientation_check_results.json`.

## 23.2 H3's twelve pentagons: zero vertices on land

H3's resolution-0 grid contains exactly twelve pentagonal cells, one centred on each vertex of the underlying icosahedron — Chapter 21's structural fact, here used as the extraction method rather than merely cited: `h3.get_pentagons(0)` returns exactly twelve cells, and each cell's centre (`h3.cell_to_latlng`) is, by construction, one of the icosahedron's twelve vertices. Checking all twelve against `global_land_mask` finds every one of them in ocean:

| Latitude | Longitude | Status |
|---|---|---|
| $64.70°$ | $10.5362°$ | ocean |
| $50.1032°$ | $-143.4785°$ | ocean |
| $39.10°$ | $122.30°$ | ocean |
| ⋮ | ⋮ | ⋮ (all twelve) |
| $-64.70°$ | $-169.4638°$ | ocean |

**Key Result**

Zero of H3's twelve base-cell vertices fall on land, checked against an independent raster from either paper's own coastline data. This alone rules out one of Chapter 21's three named candidates directly: Sahr et al. (2003)'s own historical software lineage places exactly *one* vertex on a continent (Section 21.2), and H3 places none. Whatever orientation H3 uses, it is drawn from the zero-vertices-on-land family that includes both Fuller's orientation and four of Barnes (2019)'s optimized rows, not from Sahr's own convention.

## 23.3 Not any of Barnes's four optimized rows

Barnes (2019)'s Table 2 lists four distinct zero-vertices-on-land icosahedron orientations, each optimizing a different objective (Section 21.3). Reconstructing each row's full twelve-vertex set with the verified placement algorithm and matching it against H3's twelve pentagon centres by nearest-neighbour angular distance rules out an exact match to any of them:

| Barnes (2019) row | Mean nearest dist. (°) | Max nearest dist. (°) |
|---|---|---|
| Max. avg. dist., 0 on land | 27.33 | 35.72 |
| Min. avg. dist., 0 on land | 15.02 | 18.96 |
| Max. edge overlap, 0 on land | 26.92 | 35.17 |
| Min. edge overlap, 0 on land | 6.87 | 8.60 |

The closest of the four is the min-edge-overlap row — unsurprising, since Barnes (2019) themselves identify that row as the one most closely resembling Fuller's own orientation (Section 21.3) — but at a mean nearest-vertex distance of nearly 7° (some 760 km at the equator) it is not a match; H3 does not implement any of Barnes's four published optimized orientations verbatim.

## 23.4 A match to Fuller, to within metres

Barnes (2019) report that self-consistent numeric coordinates for Fuller's own orientation did not exist in the literature before their paper (Section 21.2), so no Fuller row appears in Table 2 to test against directly. A later paper in the same lineage, Zhou et al. (2020), transcribes a primary-vertex coordinate for Fuller's orientation: $(2.3009°\text{N}, 5.2454°\text{W})$. This is, to all four published decimal places, one of H3's own twelve pentagon centres (the third row of Section 23.2's table, restored to full precision: $2.3009°, -5.2454°$) — an exact coincidence at the primary-vertex level, before any rotation parameter is even considered.

**Methodological Note**

Zhou et al. (2020) also report a rotation parameter for this orientation, $a = 7.4666°$, transcribed in their own notation rather than Barnes (2019)'s $\theta$. Reconstructing the full orientation with $\theta = 7.4666°$ at this primary vertex and matching against H3's pentagons gives a mean nearest-vertex distance of $11.79°$, far too large to be the same angle in Barnes's convention. This is disclosed rather than silently discarded: $a$ and $\theta$ are evidently different parameterizations of the rotation, not interchangeable, and this chapter does not attempt to reconcile $a$'s own definition beyond noting the mismatch.

Searching $\theta$ directly at the shared primary vertex — rather than trusting either paper's reported rotation value — for the angle that best aligns the reconstructed twelve-vertex set to H3's own finds a value under which every one of H3's twelve pentagons matches a reconstructed vertex to within $2.3$ metres, at the limit of the four-decimal input precision:

**Key Result**

H3's icosahedron orientation matches Barnes (2019)'s placement of Fuller's Dymaxion orientation, using Zhou et al.'s own published primary vertex, to within $2.3$ metres across all twelve vertices simultaneously — not a resemblance, an exact match at the precision available. Independently, an issue on H3's own GitHub repository records Barnes (2019) himself, commenting directly, stating that H3 fixes its icosahedron using "a Dymaxion orientation (due to R. Buckminster Fuller)" (Barnes, 2021). The geometric reconstruction here and this primary-source statement corroborate each other independently: one a numeric fit performed without reference to the GitHub discussion, the other a direct claim from the author of the paper this volume has otherwise only used as a source of published coordinates.

The multiple values of $\theta$ found by an unconstrained search (differing from each other by multiples of $36°$) all give the same near-metre-level match; this reflects the icosahedron's own rotational symmetry about a fixed vertex axis, under which several values of $\theta$ describe geometrically identical placements, not an ambiguity in the result itself.

## 23.5 What remains open

This chapter closes the H3 branch of Chapter 21's todobox with a positive, quantified result: H3 uses Fuller's orientation, not an independently chosen one, and not any of Barnes (2019)'s four optimized alternatives. DGGRID's default ISEA7H configuration and OpenEAGGR's default ISEA3H configuration remain unchecked, for the same reason Chapters 15 and 16 already disclosed for other questions: both require compiled binaries not available in this environment. The method used here — extract each library's base-polyhedron vertex coordinates, check them against land, and fit $\theta$ at a shared primary vertex if a zero-land candidate is found — would apply to either directly, once those binaries are available.

Chapter 24

# The Modifiable Areal Unit Problem, Rediscovered on a Grid

## 24.1 Scope: a problem imported from human geography, not derived here

Every chapter of Part I so far has treated a DGGS cell as a fixed, mathematically specified object: something produced once by a projection (Chapter 19) and an orientation (Chapter 21) and then counted, timed, or measured for area and compactness (Chapter 11 and the chapters that follow it). This chapter asks a question about that same object from an entirely different literature, one this volume has not yet drawn on at all: human geography's *Modifiable Areal Unit Problem* (MAUP), the observation that statistics computed over spatial zones are sensitive not just to the phenomenon being measured but to the — usually arbitrary — choice of zones used to measure it. The question is not whether DGGS cells are distorted (that is what Chapters 11 and 19 already establish) but whether the very fact that a DGGS's cells are mathematically fixed, rather than administratively drawn, changes anything about MAUP, or merely relocates its two component problems onto the cartographic design choices this volume has already made.

**Methodological Note**

This chapter performs no new computation and imports no new numeric values from outside sources already in Section 114.6 plus two additions introduced here (Raposo et al., 2019; Nelson & Brewer, 2017). Its only original contributions are (a) a re-reading of MAUP's classical scale/zonation distinction against a DGGS cell rather than a census zone, and (b) a re-framing, under that lens, of numeric results this volume already established in Chapters 11 and 21 without altering a single figure reported there. Nothing below should be read as a new empirical finding; it is a reclassification of existing findings under an apparatus — MAUP — that geography's social-science branch has used for over ninety years and that this volume's cartographic branch has not, until now, named explicitly.

## 24.2 The classical formulation: two problems, one cause

The effect that would become MAUP was first documented, without that name, by Gehlke & Biehl (1934): correlation coefficients computed from the same underlying census data changed, sometimes substantially, depending on how source records were grouped into census-tract units before the correlation was run. The name itself, and the two-part decomposition still used today, is due to Openshaw (1984): the *scale problem*, where aggregating the same data into progressively larger zones (e.g. tract → county → state) changes the computed statistic even when every zone boundary is nested consistently; and the *zonation problem*, where re-drawing zones of the *same* size and number, but in a different arrangement, changes

the statistic again even though nothing about the scale has changed. Openshaw's own diagnosis of the root cause — that "the areal units (zonal objects) used in many geographical studies are arbitrary, modifiable, and subject to the whims and fancies of whoever is doing, or did, the aggregating" — is the sentence this chapter keeps returning to, because a DGGS cell is, by construction, precisely the kind of object that sentence seems to rule out.

Openshaw & Taylor (1979) demonstrated the scale of the effect with an experiment on Iowa county-level data, recomputing a single correlation (percentage voting Republican against percentage of elderly residents) under a large number of alternative county aggregations; secondary literature reporting this experiment states that the resulting correlation ranged from roughly $-0.97$ to $+0.99$ across aggregations of the identical underlying records (as summarized by Wagner, 2022).

**To Complete / Verify**

The $-0.97$ to $+0.99$ figure above is taken from a secondary summary of Openshaw & Taylor (1979), not verified against the original 1979 book chapter itself, which was not accessible in this pass (it is a chapter in an edited volume, not an independently indexed journal article with its own DOI). It is reported here, flagged as secondary, only to convey the *order of magnitude* of the classical scale-problem effect, and no further numeric claim is built on it.

A closely related but logically distinct effect is the *ecological fallacy*: the error of inferring an individual-level relationship from a correlation observed only at the aggregate level. Robinson (1950) gave this its classical demonstration, again using U.S. census data, showing that a strong aggregate-level correlation between two variables need not reflect any relationship between those variables at the level of the individuals being aggregated. MAUP and the ecological fallacy share a cause — both arise from the act of aggregating point-level or individual-level data into zones — but they are not the same claim: MAUP is about the *instability* of a statistic under different choices of zone, while the ecological fallacy is about the *invalidity* of a particular cross-level inference, and a zoning scheme can be perfectly stable (no MAUP) while the ecological fallacy still applies to any inference drawn from it, or vice versa. Fotheringham & Wong (1991) extended the scale/zonation decomposition from simple bivariate correlation to multivariate regression, showing that estimated regression coefficients, not just correlation coefficients, are subject to the same two-part instability — the result that makes MAUP a standing concern for any statistical model fitted to arbitrarily aggregated spatial data, not a curiosity confined to Openshaw's original correlation experiments.

## 24.3 Does a DGGS abolish the problem, or just relocate it?

The zones in every MAUP study cited in Section 24.2 — census tracts, counties, electoral wards — are exactly the kind of object Openshaw's diagnosis targets: drawn by an administrative process, for administrative reasons, with no necessary relationship to the phenomenon later measured within them, and free to be redrawn by the next census or the next redistricting. A DGGS cell looks, at first, like the opposite of this: its shape and its position are fixed once, by a projection formula (Snyder, 1992; Gibb et al., 2016; Górski et al., 2005, all discussed in Chapter 19) and an orientation (Barnes, 2019, discussed in Chapter 21), and thereafter identical for every dataset ever binned into it — nobody redraws an H3 cell to help one candidate's vote share, and the same resolution-2 rHEALPix cell used in Chapter 11 means the same patch of the Earth's surface regardless of what is being measured inside it. It is tempting to conclude from this that a DGGS simply *solves* the zonation half of MAUP by removing the human discretion that causes it.

Raposo et al. (2019) built a pedagogical tool that shows why this conclusion is too quick, and did so using a DGGS explicitly for that purpose: an interactive globe built on Dutton's quaternary triangular mesh (QTM), designed to teach students the MAUP by letting them manipulate the tessellation's two free parameters directly. Their tool lets a user (a) select the QTM subdivision level at which point data are binned — an on-screen version of the classical scale problem, with the same set of point observations aggregated into fewer, larger cells or more, smaller cells at the user's choice — and (b) translate the whole tessellation east or west across the globe at a fixed level, which changes which points fall into which bin without changing the number, size, or shape of the bins at all — a clean, on-screen version of the zonation problem, isolated from the scale problem in a way that is difficult to demonstrate with real, irregular administrative boundaries, since a census tract cannot simply be "slid" three degrees west and remain a recognizable census tract.

**Key Result**

Raposo et al. (2019)'s translation experiment is the key fact this chapter needed from outside this volume's own material: it shows that a DGGS does not remove the zonation problem, it *isolates* it into a single free parameter (the tessellation's global placement) that can be varied independently of resolution, in a way arbitrary administrative zones cannot be. The scale problem survives entirely intact — resolution remains a free choice, exactly as in classical MAUP — and the zonation problem survives as well, relocated from "which ward boundaries were drawn" to "which orientation and which DGGS family were chosen," the two design axes Chapters 19 and 21 already treat as independent cartographic decisions.

## 24.4 Reframing this volume's own results as instances of the two problems

Two results already established elsewhere in this volume, neither previously described in MAUP's terms, turn out to be direct instances of its two components.

**The zonation problem, instance one: orientation.** Section 21.4 of Chapter 21 already showed, using Barnes (2019)'s published coordinates, that the same DGGS at the same resolution places its twelve structurally singular pentagonal cells over open ocean under one orientation and over populated landmasses under another — the identical point on the Earth's surface falls under a regular hexagonal cell in one orientation and under a smaller, structurally distinct pentagonal cell in another. This is the zonation problem in its purest form: same number of zones, same zone sizes and shapes considered as a population, different assignment of specific Earth locations to specific zones, purely as a function of a single upstream design choice (the orientation parameters $\lambda, \varphi, \theta$) that has nothing to do with the data later binned into the grid.

**The zonation problem, instance two: choice of DGGS family.** Tables 11.1 and 11.2 of Chapter 11, transcribed from Kmoch & Vasilyev (2022), already report that the same sphere, tessellated at comparable resolutions by five different DGGS constructions, yields normalized-area standard deviations spanning roughly $0.002$ (rHEALPix, resolution 2) to $0.149$ (S2, resolution 4) and mean isoperimetric quotients spanning $0.767$ to $0.905$. Any global statistic computed by binning point data into cells — a density, a rate, a mean — will differ across these five choices for the same underlying reason Openshaw and Taylor's correlation swung between $-0.97$ and $+0.99$ across alternative Iowa county aggregations: the zones differ, even though nothing about the phenomenon does. The difference from the classical case is that here the five "zonings" being compared are not five arbitrary conventions but five fully specified, independently reproducible mathematical constructions, each already documented in this volume's

own reproduction chapters (see Chapters 12, 15 and 16).

**The scale problem, instance: resolution.** The same reproduction tables record cell counts growing from 5,736 (H3, resolution 2) to 284,649 (H3, resolution 4), and Chapter 13 extends this across further levels for H3 and S2. A rate or density computed by binning the identical set of point observations at resolution 2 versus resolution 4 is subject to exactly Openshaw's scale problem, with the DGGS's recursive, hierarchical subdivision playing the role classical MAUP studies gave to the tract/county/state hierarchy of administrative units.

**To Complete / Verify**

What is *not* shown here, and would require new work outside this chapter's scope, is an actual recomputation of some statistic (a correlation, a rate) over real point data binned at these different resolutions or orientations, of the kind Openshaw & Taylor (1979) and Raposo et al. (2019) each carried out for their own zonations. This chapter reclassifies already-established structural facts about DGGS cells under the MAUP framework; it does not run a new MAUP experiment on a DGGS, and no such experiment's numeric outcome is asserted or estimated.

## 24.5 A cartographic, not only statistical, restatement of MAUP

MAUP is usually presented, as in Section 24.2, as a problem for statistical inference: a correlation or a regression coefficient changes with the zoning. Nelson & Brewer (2017), published in Cartography and Geographic Information Science rather than a statistics or social-science venue, restate the same problem in explicitly cartographic terms: using local indicators of spatial association (LISA) computed on the same variable (income, disease rates) across nested U.S. census hierarchies (county, tract, block group), they show that the *spatial pattern* revealed by a choropleth map — not merely a single summary statistic — is itself unstable across levels of aggregation, and propose a method for identifying which aggregation level is comparatively more trustworthy for a given variable rather than assuming any one census tier is automatically appropriate. This reframes MAUP as a mapmaking decision as much as a statistical one: the question is not only "does my correlation change" but "does the map I am about to draw, and the pattern a reader will see in it, survive a change of zoning."

**To Complete / Verify**

No source located in this pass applies Nelson & Brewer's LISA-based stability method to a DGGS cell hierarchy specifically, as opposed to a census hierarchy. Doing so — checking whether a variable's spatial pattern is more or less stable across H3 resolutions than across U.S. census tiers, for instance — is a natural extension flagged here as unattempted, not claimed as a result of this chapter.

## 24.6 Synthesis: relocated, not resolved

Openshaw's 1984 diagnosis was that arbitrary, modifiable zones are "subject to the whims and fancies of whoever is doing... the aggregating." A DGGS cell removes exactly that: nobody redraws an H3 cell case by case, and the same cell means the same patch of Earth for every dataset that uses it. What Sections 24.3 and 24.4 show is that this does not touch either half of MAUP's two-part decomposition — it only changes *where* the discretion that produces each half now sits. The scale problem is exactly as open as ever, now expressed as "which resolution level"

rather than "which administrative tier." The zonation problem is exactly as open as ever, now expressed as "which orientation, which base polyhedron, which of the five DGGS families already compared in Chapter 11" rather than "which ward boundaries." The one genuine gain, illustrated concretely by Raposo et al. (2019)'s translation experiment and by this volume's own orientation and multi-DGGS comparisons, is that these relocated choices are fully specified, reproducible, and independently checkable — exactly the property Section 114.6's verification discipline already demands of every other design choice in this volume, and exactly the property arbitrary census geography never had. A DGGS does not solve the Modifiable Areal Unit Problem Openshaw named forty-two years before this volume's writing; it hands the same two-part problem to cartography, in a form cartography's own tools — projection, orientation, resolution — are equipped to reason about directly.

Chapter 25

# Aperture and Hierarchy: A Third Design Axis

## 25.1 Scope: two axes fixed, a third left untouched

Chapters 19 and 21 isolated two cartographic design choices that any ISEA-family DGGS makes once, before any cell is ever counted: the projection formula that maps a polyhedron's faces to the sphere, and the orientation that places the polyhedron on the actual Earth. Both chapters left a third choice explicitly to one side: how a cell at one resolution subdivides into the cells of the next, finer resolution. This is the choice of *aperture*, and it is logically independent of the other two in exactly the sense Section 21.5 established for projection and orientation — a change to the aperture sequence changes the resulting DGGS's hierarchy without touching either the projection formula or the polyhedron's placement on the globe.

**Methodological Note**

As in Chapter 21, this chapter performs no new computation. It draws entirely on the hexagonal-DGGS design literature (Sahr et al., 2003; Sahr, 2008, 2011; Kmoch et al., 2025) and on human geography's central place theory (Christaller, 1966), which that literature explicitly borrows its terminology from — a cross-disciplinary borrowing this volume has now met twice, since Chapter 24 borrowed the Modifiable Areal Unit Problem from the same broad field for an unrelated reason. No numeric value below is estimated, interpolated, or derived from a new source; each is transcribed and attributed as noted.

## 25.2 What an aperture is, and where its name comes from

For a hexagonal DGGS, the *aperture* of a subdivision step is the ratio of a parent cell's area to a child cell's area: an aperture-$N$ subdivision replaces one hexagon with $N$ smaller hexagons at the next resolution. Sahr et al. (2003) identify three apertures that admit a subdivision preserving exact hexagonal symmetry: 3, 4, and 7, corresponding to the three smallest solutions of $N = i^2 + ij + j^2$ for positive integers $i, j$ ($i{=}1, j{=}1$ gives $N{=}3$; $i{=}2, j{=}0$ gives $N{=}4$; $i{=}2, j{=}1$ gives $N{=}7$). Sahr and collaborators call these the *central place* apertures, adopting the term directly from Christaller (1966)'s central place theory in human geography, where a central place's hexagonal market area is itself partitioned into smaller hexagonal areas served by lower-order central places. The DGGS literature's use of the same word for the same geometric relationship — a hexagon subdivided by smaller hexagons whose centres coincide with the parent hierarchy's structure — is a direct, acknowledged borrowing, not a coincidence of vocabulary.

## 25.3 Aperture 7: symmetric, but rotated

Among the three central place apertures, aperture 7 is singled out repeatedly in the literature surveyed here for preserving hexagonal symmetry most completely: Kmoch et al. (2025) state that aperture-7 subdivision preserves hexagonal symmetry and yields an unambiguous indexing hierarchy, properties the other two central place apertures do not share to the same degree, and note that this is the aperture underlying Uber's H3, already discussed at length in Chapter 12. The symmetry comes at a specific, documented cost: an aperture-7 subdivision's child-hexagon grid is rotated by approximately $19.1°$ relative to its parent's grid at every level, a figure given directly in H3's own documentation and repeated in the secondary literature surveyed alongside it. This per-level rotation is why an aperture-7 hierarchy such as H3's cannot be described by a single fixed alignment of child cells inside a parent cell: the alignment itself rotates step by step down the hierarchy.

**To Complete / Verify**

The $19.1°$ figure above is taken from H3's own project documentation and from a secondary source citing it (referenced informally in the literature surveyed here as "Bousquin 2021; Uber Technologies 2023"); neither primary source was fetched directly in this pass, so the figure is reported with the same caution Chapter 24 applied to a secondary-sourced number, and no further claim is built on it beyond stating the qualitative fact of a per-level rotation.

## 25.4 Congruent and non-congruent hierarchies

Sahr et al. (2003) give a name to the property this rotation puts out of reach for hexagons: a DGGS hierarchy is *congruent* if and only if every child cell at one resolution is entirely contained within exactly one parent cell at the coarser resolution, with no cell straddling a parent boundary. Quadrilateral- and triangle-based hierarchies (S2's quadtree subdivision, discussed in Chapter 19, or a simple triangular subdivision of an icosahedral face) can achieve exact congruency, because a square or a triangle can be partitioned into smaller, similar copies of itself with no residue. A regular hexagon cannot: no integer number of smaller regular hexagons tiles a larger regular hexagon exactly, for the same elementary tiling reason that hexagons, unlike squares and triangles, do not admit self-similar subdivision. Section 21.4 already used one consequence of this fact — the twelve unavoidable pentagonal cells at an icosahedron's vertices — without naming its cause explicitly; the cause is exactly this non-congruency. Bondaruk et al. (2020), reviewing four DGGS software implementations (dggridR, H3, OpenEAGGR, and S2) against the OGC's DGGS abstract specification, state the same structural fact directly: hexagons "cannot be refined into self-similar shapes without overlap," which is also why a non-hexagonal cell type must appear somewhere in any hexagonal DGGS built on a polyhedron with vertices.

**Key Result**

The choice of aperture is not merely a matter of how quickly resolution refines. It determines whether the resulting hierarchy is congruent (quadrilateral and triangular subdivisions, aperture 4 on those cell types) or non-congruent (any hexagonal aperture), independently of which projection and which orientation was chosen for the underlying polyhedron. A DGGS designer who fixes projection and orientation exactly as Chapters 19 and 21 describe still faces this third, separate decision, with a structural consequence —

exact parent-child containment or its absence — that neither of the other two axes controls.

## 25.5 Pure and mixed aperture sequences

A DGGS need not use the same aperture at every resolution step. Following Kmoch et al. (2025) and the Central Place Indexing literature they build on, a *pure* aperture sequence repeats one aperture at every level (aperture-7 at every step, as in H3, or aperture-3 at every step, as in the ISEA3H configuration of DGGRID already tabulated in Chapter 11), while a *mixed* aperture sequence combines different apertures at different levels — for instance alternating aperture-4 and aperture-3 steps — to reach a wider range of target resolutions than a pure sequence of either aperture alone permits. Kmoch et al. (2025) present IGEO7 as a pure aperture-7 hexagonal DGGS built specifically to close a gap they identify in the existing open-source landscape: no open, equal-area, aperture-7 hexagonal DGGS with indexing capability comparable to H3 existed before it, since H3 itself is not equal-area. The same source states directly that H3's cell sizes vary by as much as $\pm 50\%$ across the globe, a figure that sits alongside, and is consistent in direction with, the normalized-area spread this volume already measured for H3 from Kmoch & Vasilyev (2022)'s independent data in Table 11.1.

**To Complete / Verify**

Whether the aperture-4/aperture-3 mixed sequence mentioned above, or any other specific mixed sequence, is used by a released, production DGGS other than the Central Place Indexing examples described in the sources cited here has not been checked independently in this volume; this chapter reports what the aperture-design literature states about mixed sequences in general, not a survey of which production systems use one.

## 25.6 Synthesis: a third axis, and where it leaves this volume's map

Chapters 19, 21 and 24 already established that a DGGS's cartographic behaviour rests on several logically independent design choices, each fixed once and inherited by every resolution built on top of it. Aperture is a fourth item on that list, alongside projection, orientation, and the scale/zonation pair Chapter 24 discusses, and it is specific to hierarchical, multiresolution constructions in a way the other three are not: a single-resolution grid has a projection and an orientation but no aperture, since aperture describes the relationship *between* resolutions, not a property of any one resolution alone. Every DGGS this volume has examined empirically — H3 (Chapter 12), S2, rHEALPix, DGGRID (Chapter 15), and OpenEAGGR (Chapter 16) — has made this fourth choice as well as the other three, and Section 25.4's congruent/non-congruent distinction offers a structural explanation, independent of any timing benchmark in Part III, for why the hexagon-based schemes among them share the twelve-pentagon anomaly those chapters each found empirically, while a purely quadrilateral or triangular scheme would not be structurally forced to.

Chapter 26

# A Standard Written After the Fact: The OGC DGGS Abstract Specification

## 26.1 Scope: naming a category this volume has assumed all along

Chapter 25 cited Bondaruk et al. (2020) in a single sentence, for one structural fact about hexagonal aperture-3 subdivision, and left the standard that sentence referred to otherwise unexamined. This chapter, in the institutional register Chapter 57 used for patent history, examines that standard directly: the Open Geospatial Consortium's 2017 Discrete Global Grid Systems Abstract Specification, the document that first gave the category this volume has used informally throughout — "DGGS" — a single, formally agreed definition, years after H3, S2, rHEALPix, DGGRID, HTM, and OpenEAGGR were already independently in production or research use.

**Methodological Note**

This chapter introduces two new sources to Section 114.6: OGC (2017), the standard itself (cited via its search-indexed abstract and via the secondary accounts below, since the full document text was not independently fetched in this pass), and Purss et al. (2016), an earlier conference paper by the standard's own editorial group stating its core definitions ahead of formal approval. Bondaruk et al. (2020), already in Section 114.6 for Chapter 25, is used again here at greater length, for its systematic evaluation of four DGGS implementations against the standard's own criteria. No computation is performed; every requirement and every compliance claim below is transcribed, not tested independently against source code.

## 26.2 What the standard actually requires

Purss et al. (2016) state the OGC's own definition directly: a Discrete Global Grid System is *"a spatial reference system that uses a hierarchical tessellation of cells to partition and address the globe,"* characterized by "the properties of their cell structure, geo-encoding, quantization strategy and associated mathematical functions." Read against the individual design axes this volume has already isolated one at a time — projection (Chapter 19), polyhedron orientation (Chapter 21), and aperture (Chapter 25) — the standard's definition is a retroactive formalization of exactly that same list, plus three functional requirements this volume has so far treated only implicitly: OGC (2017), per the secondary accounts consulted here, requires (i) a genuinely *hierarchical* tessellation, ruling out any single-resolution grid regardless of cell shape; (ii) *equal-area* cells at the Core Reference System level, not merely approximately equal, so that statistical aggregation across cells is meaningful without an area-weighting correction; and (iii) a defined set of *quantization*, zonal-query, and interoperability operations, so that data can be assigned to cells, cells can be related to their neighbours and to cells at other resolutions, and

different DGGS implementations can exchange results, purely from cell identifiers, without recomputing geometry.

**To Complete / Verify**

**Resolved.** The exact formal wording of the equal-area and centroid requirements had been transcribed here only from secondary sources (Purss et al., 2016 and a duplicative Geoscience Australia slide deck), not from OGC (2017)'s own text. This revision fetches the primary document directly (OGC 15-104r5, docs.ogc.org) and can now quote its own requirement wording verbatim. **Requirement 8** (equal-area cells) reads: "For each successive level of grid refinement, and for each cell geometry, a DGGS specification SHALL define DGGS Cells that are equal area (or iteratively and predictably converge on equal area) within the specified level of precision." This confirms the secondary sources' account that equal-area is a Core requirement, not merely a recommended extension, and additionally reveals a precision-based escape clause this chapter had not previously reported: the standard explicitly permits cells that only *converge* on equal area iteratively, governed by a separate, self-defined "equal area precision" (Requirement 7), rather than requiring exact equality outright. **Requirement 13** (centroid referencing) reads: "A DGGS specification SHALL define the location of a DGGS Cell reference to be the centroid of each cell," with the standard's own rationale text (Sec. 6.2.6.1) explaining that the centroid is required specifically because it is the only reference point computable by one shape-agnostic algorithm regardless of cell geometry or rotation under refinement, unlike a fixed vertex convention. Both quotations are now sourced to the standard's own document text rather than to a secondary paraphrase.

## 26.3 The four-implementation audit: mostly non-compliant, by the standard's own criteria

Bondaruk et al. (2020) evaluate four DGGS software packages — DGGRID (dggridR), H3, OpenEAGGR, and S2, three of which this volume has already examined computationally in Chapters 12, 15 and 16 — directly against the OGC's Topic 21 criteria, and report that the implementations *"generally fall short of fulfilling all of the OGC requirements or it was not possible to confirm their compliance."* This is not a marginal finding for at least one scheme this volume has already quantified independently: Kmoch et al. (2025), already cited in Chapter 25, state that H3's own cell sizes vary by as much as $\pm 50\%$ across the globe, which on the equal-area requirement just stated is not a rounding tolerance but an outright failure of a Core Reference System criterion, for the single DGGS implementation this volume's earlier chapters found to be the fastest in every operational-timing benchmark run (Chapters 93 and 94).

**Key Result**

Put beside Part III's efficiency/accuracy trade-off tables, this is the same trade-off restated in institutional rather than computational terms: H3's timing advantage and its equal-area shortfall are not two independent facts about the same scheme, but two sides of one design decision, and the OGC's own 2017 standard — three years after H3's own 2018 public release, per the timeline already established in Chapter 12 — is the first document to make failing that criterion an explicit, named non-compliance rather than an implementation detail left for each user to discover independently. rHEALPix and S2, both explicitly equal-area by construction (Chapter 19), do not share this specific shortfall, though Bondaruk et al. (2020)'s overall finding — that none of the four audited implementations fully confirms

compliance — indicates the standard's other requirements (interoperability and quantization operations, in particular) remain unevenly met even by the equal-area schemes.

## 26.4 A standard written after its own subject matter existed

This chapter's closing point is the same one Chapter 57 made about patents, transposed from proprietary claims to standards bodies: OGC (2017) was approved by OGC membership vote and announced on 24 October 2017, a full definitional consolidation of a field whose leading implementations — Sahr's ISEA3H/DGGRID lineage, OpenEAGGR, Google's S2, and, the following year, Uber's H3 — were independently designed, released, and in several cases already in production use before any single agreed specification existed to check them against. Chapter 25 already showed that one specific structural fact (hexagons cannot subdivide congruently) was independently rediscovered across the hexagonal-DGGS literature before being named as an OGC-relevant criterion; this chapter's audit shows the same retroactive relationship holding at the level of an entire implementation's headline design choice, not merely one geometric fact about it. Read together with Chapter 57's patent history, the two institutional chapters describe two different, non-competing ways this volume's central subject — the cost of representing a continuous sphere discretely — has been formalized only after independent inventors had already paid that cost, once as a proprietary claim and once as a standards body's compliance criterion.

Chapter 27

# The Auxiliary Sphere Behind Every Global DGGS

## 27.1 Scope: a step taken before the projection is ever applied

Chapter 19 opened with the projection formula that sends a polyhedron's faces to "the sphere." That phrase was left unexamined there, and in every chapter of Part I since: every DGGS this volume has discussed — H3, S2, rHEALPix, DGGRID, OpenEAGGR — is, at bottom, a construction on a perfect mathematical sphere, while the Earth these systems are built to index is not a sphere but an oblate ellipsoid of revolution, flattened at the poles by about one part in three hundred. This chapter is about the step that bridges the two: the choice of an *auxiliary sphere*, substituted for the true reference ellipsoid before any DGGS-specific projection or tessellation is applied at all. This is the volume's title made literal — "Global Grids, Auxiliary Spheres" — at the one point in the argument where an actual auxiliary sphere, in the classical geodetic sense of that term, is doing the work.

**Methodological Note**

This chapter performs no new computation. Its geodetic content is drawn from the standard reference on auxiliary latitudes and auxiliary spheres, Snyder (1987), and from Karney (2023)'s recent survey of the same material; its DGGS-specific content draws on Sahr (2026)'s own published documentation. The one numeric value quoted for the WGS84 authalic sphere's radius is taken from a worked example published under the EPSG Geodetic Parameter Dataset's documentation of the Equal Earth projection method, not derived here.

## 27.2 Why a sphere at all: the auxiliary-latitude idea

Snyder (1987)'s *Map Projections: A Working Manual*, the standard reference for this material, explains why cartography routinely inserts a sphere between the ellipsoid and the map: formulas for projecting an ellipsoid directly onto a plane are, in general, considerably more complex than the corresponding spherical formulas, so it is common practice to first map the ellipsoid onto an auxiliary sphere using one of several *auxiliary latitudes*, and only then apply an ordinary spherical projection formula to that sphere. Karney (2023)'s recent survey lists the auxiliary latitudes in standard use — geocentric, reduced (or parametric), conformal, rectifying, authalic, and isometric — each defined so that a specific geometric property of the ellipsoid (angles, areas, meridian arc length, or a monotonic relationship with geocentric latitude) is preserved exactly on the corresponding auxiliary sphere, even though no single sphere can preserve every such property of an ellipsoid simultaneously.

Two of these auxiliary spheres matter most for the DGGS constructions this volume has examined. The *conformal sphere* preserves angles: a shape on the ellipsoid and its image on the

conformal sphere agree locally in angle and hence in shape, at the cost of area. The *authalic sphere* preserves total surface area: the authalic sphere's area equals the ellipsoid's area exactly, at the cost of angular fidelity. Since every DGGS discussed in this volume advertises some form of area preservation as a design goal — and Chapter 11's own reproduced data (Kmoch & Vasilyev, 2022) reports normalized-area statistics precisely because area preservation is the property under scrutiny — the authalic sphere, not the conformal sphere, is the auxiliary sphere a global, area-oriented DGGS has reason to adopt.

## 27.3 The authalic sphere, concretely

The authalic sphere's radius, $R_A$, is defined so that a sphere of that radius has exactly the same total surface area as the reference ellipsoid. For the WGS84 ellipsoid (equatorial radius $a = 6{,}378{,}137.0$ m, inverse flattening $1/f = 298.257223563$, both already used without derivation in Chapter 19's discussion of S2's structural description), a worked example published in the EPSG Geodetic Parameter Dataset's documentation of the Equal Earth projection method gives the authalic radius as $R_A \approx 6{,}371{,}007.181$ m — smaller than the equatorial radius $a$ by about $7{,}130$ m, or roughly $0.11\%$, and larger than the polar radius $b \approx 6{,}356{,}752.314$ m by about $14{,}255$ m, sitting between the two exactly as a single averaging radius must.

**Key Result**

The authalic sphere is not an approximation invented for DGGS purposes; it is a fixed, general-purpose geodetic construction, applicable to any ellipsoid, that predates every DGGS discussed in this volume. What a global, equal-area DGGS does is choose this pre-existing construction, rather than some other auxiliary sphere, as the surface its own polyhedral tessellation and projection formula (Chapter 19) are subsequently applied to. The DGGS-specific design choices this volume has examined — projection, orientation, aperture — all operate *after* this prior step, on the authalic sphere's surface, not on the WGS84 ellipsoid directly.

## 27.4 Documented practice: DGGRID's own stated construction

This is not a reconstruction from first principles of what a DGGS "must" do; it is documented practice. Sahr (2026)'s own published user documentation, already relied on in Chapter 15 for this volume's independent DGGRID reproduction, states for at least one of its standard grid definitions that cells are projected to longitude and latitude "on a sphere with the authalic WGS84 radius (NAD83 datum)," using an inverse icosahedral projection descending from Snyder (1992)'s equal-area polyhedral projection, already discussed in Chapter 19. This single documented sentence closes the loop this chapter opened: the authalic sphere of Section 27.3 is not merely the theoretically appropriate choice for an equal-area DGGS, it is the sphere DGGRID's own documentation names explicitly as the surface its projection is applied to.

**To Complete / Verify**

Whether H3, S2, rHEALPix, and OpenEAGGR — the other four DGGS this volume has examined empirically — each document their own choice of auxiliary sphere (authalic, or some other convention) as explicitly as Sahr (2026)'s documentation does for DGGRID has not been checked here. This chapter reports one documented case, not a five-way comparison; extending the check to the other four implementations' published documentation

is left open.

## 27.5 A cautionary counter-example: when the auxiliary sphere is not authalic

Not every spherical approximation of the Earth used in mapping practice is the authalic sphere, and the distinction matters. Web Mercator, the projection underlying most consumer web maps, is documented (in ArcGIS's own published projection reference, consulted for this comparison) as using a sphere whose radius equals the WGS84 ellipsoid's *equatorial* radius, $a = 6{,}378{,}137.0$ m exactly — not the authalic radius $R_A \approx 6{,}371{,}007.181$ m of Section 27.3. This is a documented, deliberate simplification for a projection that was never claimed to be equal-area in the first place (Mercator's defining property is conformality, not authalicity), but it is worth stating plainly as a contrast: substituting *any* sphere for the ellipsoid is not the same design choice as substituting the *authalic* sphere specifically, and a reader moving from a global DGGS's documentation to a web-mapping projection's documentation should not assume the word "sphere" means the same radius in both places.

**Key Result**

The authalic sphere is the auxiliary sphere appropriate to an area-preserving construction specifically; a conformal or a merely convenient sphere (such as Web Mercator's equatorial-radius sphere) is the right choice, or an accepted simplification, for a different design goal. Section 27.4's documented DGGRID case shows a DGGS made explicit that it uses the geodetically correct sphere for its own stated equal-area goal; Section 27.5's Web Mercator case shows a widely used projection that documents using a different sphere entirely, appropriately for its own different goal. Neither is an error; each is a different instance of the same underlying design choice this chapter has isolated.

## 27.6 Synthesis: the sphere before the sphere

Chapters 19, 21 and 25 each isolated a cartographic design choice made once, upstream, and inherited by every resolution of a DGGS built on top of it: which projection, which orientation, which aperture sequence. This chapter adds a choice that sits even further upstream than all three, logically prior to the very idea of a polyhedron inscribed in "the sphere": which sphere. For an equal-area DGGS the geodetically motivated answer is the authalic sphere, a construction older than and external to DGGS design entirely, documented in Snyder (1987) decades before Sahr et al. (2003)'s geodesic DGGS framework and confirmed, in at least one documented case (Sahr, 2026), to be the sphere a real, already-examined DGGS implementation names as its own foundation.

Chapter 28

# The Auxiliary Latitudes, in Formula

## 28.1 Scope: from prose to equation

Chapter 27 named the conformal and authalic spheres and explained, in prose, why a global DGGS has reason to prefer the authalic one. Chapter 75 named two more — the rectifying sphere and the parametric sphere — and traced them to Adams's 1921 catalogue, but did not write down what distinguishes one from another beyond the geometric property each is built to preserve. This chapter closes that gap: it states, as equations rather than descriptions, the defining relations for all four classical auxiliary latitudes together with the two auxiliary latitudes that the earlier chapters left unnamed — the geocentric and isometric latitudes — and shows which of the six admit a closed form and which do not.

**Methodological Note**

This chapter performs no new computation and introduces no new source. Every equation below is transcribed, with notation lightly adapted for consistency with the rest of this volume, from Karney (2023), already cited in Chapter 27; Karney (2023) in turn attributes the underlying results to Tissot (1881)'s contemporaries Bessel (1825) and Adams (1921), and to Krüger (1912) for the series linking the conformal and rectifying latitudes, none of which are fetched directly here. No numeric value is evaluated in this chapter; the one worked numeric example in this volume's treatment of auxiliary spheres remains the WGS84 authalic radius already given in Chapter 27.

## 28.2 Six latitudes, one ellipsoid

Fix a reference ellipsoid of revolution with equatorial radius $a$ and polar semi-axis $b$, flattening $f = (a - b)/a$, and eccentricity $e$ with $e^2 = 1 - (b/a)^2$. For a point at geographic (geodetic) latitude $\phi$ — the latitude an ordinary GPS receiver reports, defined as the angle between the ellipsoid's surface normal and the equatorial plane — Karney (2023) tabulates six auxiliary latitudes, each obtained from $\phi$ by a different rule and each preserving, by construction, a different geometric quantity when the ellipsoid is replaced by a sphere.

**Parametric (reduced) latitude $\beta$.** Stretch the ellipse of the meridian into a circle along the polar axis; the parametric latitude is the angle on that circle corresponding to $\phi$ on the ellipse,

$$\beta = \tan^{-1}\big((1 - f)\tan\phi\big).$$

**Geocentric latitude $\theta$.** The angle between the equatorial plane and the line from the ellipsoid's centre (not its surface normal) to the point,

$$\theta = \tan^{-1}\big((1 - f)^2\tan\phi\big).$$

**Rectifying latitude $\mu$.** Defined so that equal increments of $\mu$ correspond to equal distances along the meridian. Writing $s$ for the meridian arc length measured from the equator to latitude $\phi$ and $s_p$ for the quarter meridian (the arc length from equator to pole),

$$\mu = \frac{\pi}{2}\,\frac{s}{s_p}, \qquad s = a\left(E(\phi, e) - \frac{e^2 \sin\phi\cos\phi}{\sqrt{1 - e^2\sin^2\phi}}\right),$$

with $E(\phi, e)$ the incomplete elliptic integral of the second kind.

**Conformal latitude $\chi$.** Defined so that the mapping $\phi \mapsto \chi$ is angle-preserving; equivalently, $\chi = \mathrm{gd}\,\psi$ where $\psi$ is the isometric latitude below and gd is the Gudermannian function.

**Authalic latitude $\xi$.** Defined so that the area between the equator and the circle of latitude $\phi$ on the ellipsoid equals the area between the equator and latitude $\xi$ on the sphere,

$$\xi = \sin^{-1}\frac{A(\phi)}{A_p}, \qquad A(\phi) = \pi b^2\left(\frac{\tanh^{-1}(e\sin\phi)}{e} + \frac{\sin\phi}{1 - e^2\sin^2\phi}\right),$$

with $A_p$ the value of $A$ at the pole (half the ellipsoid's total surface area).

**Isometric latitude $\psi$.** Not itself one of the four spheres this volume names, but the quantity every conformal construction routes through: the vertical coordinate of the ellipsoidal Mercator projection,

$$\psi = \mathrm{gd}^{-1}\,\phi - e\tanh^{-1}(e\sin\phi).$$

Unlike the other five, $\psi$ diverges at the poles rather than approaching $\pm\pi/2$, which is why Karney (2023) does not count it among the auxiliary latitudes proper even though every conformal-latitude computation passes through it.

**Key Result**

Four of these six latitudes correspond to the four auxiliary spheres named across this volume: $\beta$ and $\theta$ are used implicitly wherever this volume treats geodesic problems on "the" sphere without further qualification; $\mu$ is Adams's rectifying sphere of Section 75.2, the one Kerkovits situates his own low-distortion regional sphere against; $\xi$ is the authalic sphere of Section 27.3, the one this volume's global, area-oriented DGGS chapters have reason to prefer. The conformal latitude $\chi$ names the conformal sphere of Section 27.2, still in active national use per Chapter 75, but disfavoured for area-oriented DGGS work for exactly the reason Chapter 27 gave: it buys angular fidelity by giving up area.

## 28.3 Which of the six close, and which do not

The six defining relations above divide sharply by how far they are from an elementary function. $\beta(\phi)$ and $\theta(\phi)$ (Section 28.2) are elementary and trivially invertible: both $\tan\beta$ and $\tan\theta$ are $\phi$'s own tangent rescaled by a constant factor depending only on $f$, so $\phi$ recovers from either by the same rescaling in reverse. The remaining three — $\mu$, $\chi$, $\xi$ — each involve either an elliptic integral ($\mu$, via $s$) or a transcendental combination of $\sin\phi$ and $\tanh^{-1}(e\sin\phi)$ ($\chi$ via $\psi$, and $\xi$ via $A$), and Karney (2023) states plainly that none of the three can be inverted in closed form: recovering $\phi$ from a given $\mu$, $\chi$, or $\xi$ requires either a truncated series expansion in the ellipsoid's third flattening $n = (a-b)/(a+b)$, convergent for ordinary terrestrial eccentricities, or Newton's method applied to the defining equation directly.

**To Complete / Verify**

Karney (2023)'s own series expansions (the matrices $C_{\eta\zeta}$ of that paper's Appendix A) are the modern, double-precision-accurate replacement for the lower-order series Adams (1921) and Snyder (Snyder, 1987) give for the same conversions, but transcribing those matrices here would add machinery this volume does not otherwise use (this volume's own numeric work is confined to the DGGS chapters of Parts I and III and does not include ellipsoid-to-sphere latitude conversion at any point). They are left uncited beyond this note; a reader who needs the conversions themselves is directed to Karney (2023) directly or to the open-source GeographicLib library the same author maintains.

## 28.4 Why this matters for a volume about discrete grids

None of the six latitudes above is specific to DGGS; all six predate digital cartography by decades (Bessel's parametric latitude, 1825) or more than a century (Adams's systematic catalogue, 1921). What ties this material to the rest of the volume is the same point Chapter 75 already made about the rectifying sphere specifically: every one of these constructions substitutes a *sphere* for the true reference ellipsoid, and every DGGS this volume has examined — H3, S2, rHEALPix, DGGRID, OpenEAGGR alike — inherits whichever substitution its designers chose, silently, at the very first step, before any hexagon or triangle is drawn. The six formulas of Section 28.2 are the full menu that step was ever free to choose from; Chapters 27 and 75 identified which two items on that menu (authalic, rectifying) this volume's DGGS and regional-map discussions respectively depend on, and this chapter has now written both of them down, alongside the four alternatives that were available but not taken.

Chapter 29

# The Auxiliary Latitudes, Evaluated: How Much the Substitution Costs

## 29.1 Scope: from equation to number, and to a Tissot of the step itself

Chapter 28 wrote down all six auxiliary latitudes as equations and stated plainly that it evaluated none of them: "No numeric value is evaluated in this chapter; the one worked numeric example in this volume's treatment of auxiliary spheres remains the WGS84 authalic radius already given in Chapter 27." Chapter 85 separately named a second, more specific gap: the six defining formulas give the auxiliary *latitude* at each point, but "no chapter has ever measured how much it costs, in local distortion, the step ellipsoid→autalic sphere – the only link of the entire volume's pipeline never subjected to a real Tissot." This chapter closes both: it evaluates the six formulas numerically, and it computes a genuine Tissot indicatrix – scale factors $h, k$ and angular deformation $\omega$ – for the pure ellipsoid-to-sphere reassignment step, for the three auxiliary spheres this volume actually uses elsewhere (authalic, rectifying, conformal), before any further DGGS-specific or regional-map-specific projection is applied on top of it.

**Methodological Note**

**What was done, precisely.** All six auxiliary latitudes of Section 28.2 were coded independently from their defining equations (the four elementary or transcendental closed forms directly; the rectifying latitude $\mu$ via `scipy.special.ellipeinc`, the normalized incomplete elliptic integral of the second kind, matching Karney (2023)'s own definition), all on the WGS84 ellipsoid ($a = 6{,}378{,}137.0$ m, $1/f = 298.257223563$) already used throughout this volume. The authalic latitude $\xi$ was cross-checked against `rhealpixdggs.utils.auth_lat`, the same library function already exercised by Chapters 31 and 33's ellipsoidal rHEALPix pipeline: the two independent implementations agree to better than $10^{-9}$ arcseconds at every one of nineteen latitudes checked, from $0°$ to $90°$ in $5°$ steps.
The three sphere radii used in Section 29.3 — the rectifying radius $R_{\text{rect}} = 6{,}367{,}449.146$ m, the authalic radius $R_A = 6{,}371{,}007.181$ m, and the conformal radius $R = a = 6{,}378{,}137.000$ m — are not newly chosen here; they are the same three conventions already fixed and disclosed in `code/kerkovits_vs_traditional_spheres.py` for Section 76.4.3, and $R_{\text{rect}}$ is independently reproduced here from the elliptic integral directly ($R_{\text{rect}} = (2/\pi)s_p$, $s_p$ the quarter meridian) as a cross-check against that earlier figure: the two agree to the millimetre.
Because each substitution $\phi \mapsto \{\xi, \mu, \chi\}(\phi)$ depends on latitude alone – longitude is carried through unchanged – the Jacobian of the reassignment step is exactly diagonal at every point: there is no shear term, $\sin\theta' = 1$ exactly, and the Tissot semi-axes are simply the meridional scale factor $h = R\,\mathrm{d}(\text{aux. lat.})/(M(\phi)\,\mathrm{d}\phi)$ and the parallel scale factor $h = R\cos(\text{aux. lat.})/(N(\phi)\cos\phi)$, with $M, N$ the WGS84 meridional and prime-vertical radii of curvature already used in Chapter 33. This is a genuine Tis-

sot computation – not a simplification of one – for the special case of a diagonal Jacobian; it needed no general finite-difference machinery of the kind Chapter 31 used for grids whose Jacobian is not diagonal. Derivatives were evaluated by central difference at $\varepsilon = 10^{-6}$ rad. Full code: `code/auxiliary_latitude_tissot.py`; full output: `data/auxiliary_latitude_tissot_results.json`.

## 29.2 The six latitudes, evaluated: how far the substitution moves a point

Table 29.1: Latitude reduction $\phi -$ (aux. latitude), WGS84, arcminutes. $\beta$ parametric, $\theta$ geocentric, $\mu$ rectifying, $\chi$ conformal, $\xi$ authalic. Source: `code/auxiliary_latitude_tissot.py`.

| $\phi$ (°) | $\beta$ | $\theta$ | $\mu$ | $\chi$ | $\xi$ |
|---|---|---|---|---|---|
| 0 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 15 | 2.882 | 5.756 | 4.322 | 5.756 | 3.843 |
| 30 | 4.995 | 9.982 | 7.491 | 9.979 | 6.660 |
| 45 | 5.773 | 11.545 | 8.659 | 11.539 | 7.698 |
| 60 | 5.004 | 10.015 | 7.507 | 10.007 | 6.673 |
| 75 | 2.891 | 5.790 | 4.337 | 5.783 | 3.855 |
| 90 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |

**Key Result**

All six reductions peak near $\phi \approx 45°$ and vanish at both the equator and the pole, as the algebra of Section 28.2 requires ($\sin 2\phi$ is the leading term in every series). The authalic reduction's own peak was located numerically at $\phi = 45.05°$, 7.698 arcminutes – a value with an independent check available in the wider geodetic literature: the geocentric reduction's peak of $11.545'$ at $45°$ matches the commonly cited figure for the geodetic-versus-geocentric latitude difference on WGS84, and this chapter's own $\theta$ column reproduces it as a byproduct of the same code, not a separately fetched number. The ordering $\theta > \chi > \mu > \xi > \beta$ holds at every interior latitude checked, consistent with each latitude's own defining geometric property: $\theta$ substitutes the ellipsoid's normal direction most aggressively (it moves furthest from $\phi$), $\beta$ least, and the two this volume actually builds spheres from, $\mu$ and $\xi$, sit between the two extremes without coinciding with either.

## 29.3 The step itself, in local distortion: authalic, rectifying, conformal

**Three internal consistency checks, not just one.** Every entry in Table 29.2 satisfies a definitional identity that serves as its own check, independent of the finite-difference code: (i) the rectifying sphere's $h$ column is 1.00000 at all fifteen latitudes sampled (only seven shown) – exactly as it must be, since the rectifying latitude is *defined* by equal meridian-arclength increments, so its meridional scale factor is identically 1 by construction, not merely close to 1 within numerical tolerance; (ii) the conformal sphere's $h$ and $k$ columns agree

Table 29.2: Tissot indicatrix of the ellipsoid-to-sphere reassignment step alone, WGS84 → each sphere, before any further projection. $h$ meridional scale, $k$ parallel scale, $\omega$ maximum angular deformation. Source: `code/auxiliary_latitude_tissot.py`, `data/auxiliary_latitude_tissot_results.json`.

| | authalic ($R_A$) | | | rectifying ($R_{\text{rect}}$) | | | conformal ($a$) | | |
|---|---|---|---|---|---|---|---|---|---|
| $\phi$ (°) | $h$ | $k$ | $\omega$ (°) | $h$ | $k$ | $\omega$ (°) | $h$ | $k$ | $\omega$ (°) |
| 0 | 1.00112 | 0.99888 | 0.1281.00000 | 0.99832 | 0.0961.00000 | 1.00000 | 0.000 | | |
| 15 | 1.00104 | 0.99896 | 0.1201.00000 | 0.99844 | 0.0901.00022 | 1.00022 | 0.000 | | |
| 30 | 1.00084 | 0.99916 | 0.0961.00000 | 0.99874 | 0.0721.00083 | 1.00083 | 0.000 | | |
| 45 | 1.00056 | 0.99944 | 0.0641.00000 | 0.99916 | 0.0481.00167 | 1.00167 | 0.000 | | |
| 60 | 1.00028 | 0.99972 | 0.0321.00000 | 0.99958 | 0.0241.00251 | 1.00251 | 0.000 | | |
| 75 | 1.00008 | 0.99992 | 0.0091.00000 | 0.99989 | 0.0061.00313 | 1.00313 | 0.000 | | |
| 89 | 1.00000 | 1.00000 | 0.0001.00000 | 1.00000 | 0.0001.00336 | 1.00336 | 0.000 | | |

to at least eight significant figures at every latitude, and $\omega$ is $\mathcal{O}(10^{-8})^\circ$ throughout – angle preservation, the conformal latitude's defining property, recovered numerically rather than assumed; (iii) the authalic step's own area-scale factor $hk$ (not tabulated; in the underlying JSON) is 1 to nine significant figures at every latitude, the ellipsoid-to-sphere half of the same identity already confirmed for the full rHEALPix pipeline in Chapter 33.

**Key Result**

The authalic step alone – isolated from any subsequent DGGS projection – carries real angular distortion, up to $0.128^\circ$ at the equator, monotonically decreasing to zero at the pole; this is the number Chapter 85's todobox asked for and no earlier chapter supplied. It is small relative to rHEALPix's own subsequent angular distortion (up to $22.8^\circ$ within the polar cap, Table 33.2) but not zero, and its sign and magnitude at $\phi = 0^\circ$ ($+0.128^\circ$) matches, to three significant figures, the equator-latitude endpoint of the full-pipeline $\Delta\omega$ already measured in Chapter 33 ($+0.128^\circ$ there also) – consistent with the equatorial belt's own map being a function of latitude alone (Chapter 31), so that at the equator specifically, essentially all of the ellipsoid-versus-sphere effect on rHEALPix's $\omega$ is attributable to this one step, before any further composition. Away from the equator the two figures diverge, because rHEALPix's own subsequent map is not the identity on angular distortion and the composition of two distortion fields is not generally the sum of their $\omega$ values. The rectifying step trades its exact meridional fidelity for a comparatively larger parallel compression than the authalic step shows ($k = 0.99832$ versus $k = 0.99888$ at the equator); the conformal step, exactly angle preserving as required, instead accumulates linear scale growth away from the equator ($h = k$ rising from 1 to 1.00336 at $\phi = 89^\circ$) with no shape cost at all – the same trade Chapter 27 already stated in prose, now given a number.

**To Complete / Verify**

This closes Chapter 28's own disclaimer and Chapter 85's todobox for the three spheres this volume actually uses. The parametric and geocentric latitudes' own step-distortion is not computed here, since no chapter in this volume builds a sphere from either; Table 29.1's $\beta, \theta$ columns remain latitude-reduction figures only, not Tissot fields. Karney's own higher-order series expansions (Section 28.3's todobox) remain untranscribed for the same reason given there: this volume's own numeric work does not otherwise need them, direct evaluation via `scipy.special.ellipeinc` and the closed forms of Section 28.2 hav-

ing sufficed for every figure in this chapter.

Chapter 30

# Tissot's Indicatrix Revisited: From Cartographic Ellipse to Bi-Lipschitz Constant

## 30.1 Scope: an object cited, never examined

The Tissot indicatrix has been present in this volume since its opening page, invoked in Chapter 1 as the standard against which the whole continuous-versus-discrete argument is set: distortion in Monographs 1–4 was measured either locally, by the indicatrix, or on a finite scale, by the EVT boundary laws. That invocation treated the indicatrix as a known, settled tool and moved on. This chapter turns back to look at the tool itself — who built it, what it was built to answer, and what happened to the mathematics inside it after Tissot himself stopped writing about it. The reason to do so here, in a volume about *discrete* grids, is that the answer turns out to reconnect directly to the bi-Lipschitz apparatus this book's discrete chapters depend on, via a route that has nothing to do with DGGS and everything to do with a French army engineer's 1859 notes to the Académie des Sciences.

**Methodological Note**

This chapter introduces four new sources to Section 114.6: Tissot, 1881 (the primary object of study, cited via converging bibliographic detail from several independent secondary sources rather than fetched directly, as no open-access digitisation was located in this pass); Papadopoulos, 2017 and Miyachi et al., 2025, both fetched or cross-confirmed in this pass; and Milnor, 1969, cited only via its description in Miyachi et al., 2025, not fetched directly. No new computation is performed. Every mathematical expression reproduced below (the bi-Lipschitz constants of Section 30.4) is transcribed from Miyachi et al. (2025), itself building on Milnor (1969), and not derived independently in this volume.

## 30.2 Tissot's own construction, 1859–1881

Nicolas-Auguste Tissot (1824–1897) trained as a military engineer before becoming an instructor in geodesy; his results on map distortion were first announced in two notes to the Académie des Sciences in 1859, and only received their full, mature exposition more than twenty years later, in the 1881 *Mémoire sur la représentation des surfaces et les projections des cartes géographiques* (Tissot, 1881; Papadopoulos, 2017). The construction itself is simple to state and was, at the time, new: take an infinitesimally small circle centred at a point on the sphere, push it forward through a (differentiable) map projection, and ask what shape results on the plane. Tissot showed that the image is, in general, an ellipse, and that the two axes of that ellipse identify the two directions at the point along which the projection's local scale factor is respectively maximal and minimal (Papadopoulos, 2017). A map is conformal at a point exactly when the

indicatrix there degenerates back into a circle (possibly rescaled); it is equal-area at a point exactly when the ellipse's area matches that of the source circle regardless of its eccentricity. Every subsequent generation of cartography textbooks reproduces this device, and Chapter 1 of this volume used it exactly this way, as a settled fact requiring no further comment.

What is less often reproduced is why Tissot needed a mathematical, and not merely descriptive, account of the ellipse at all: his own research program, presented across the 1859 notes and the 1881 Mémoire, was to determine, for a region of prescribed shape, which projection minimises distortion over that region in a precise quantitative sense (Papadopoulos, 2017). That is, Tissot was already asking a version of the optimisation question that Chapter 19 of this volume associates with much later authors — Snyder's equal-area constructions, Kerkovits's minimal-distortion auxiliary spheres discussed in Part II — a century before either.

## 30.3 The quasiconformal afterlife

The indicatrix's second life began in complex analysis rather than geography. Papadopoulos (2017) traces how Tissot's treatment of the ratio between an ellipse's major and minor axes — what a cartographer reads as "angular distortion at a point" — is, in modern language, exactly the pointwise dilatation of a quasiconformal mapping. Oswald Teichmüller himself credited this lineage explicitly: in §33 of his 1940 paper on extremal quasiconformal mappings and quadratic differentials, Teichmüller identifies the origin of the quasiconformal-mapping concept in Tissot's cartographic work, a genealogy independently reported by Miyachi et al. (2025) via a direct citation of that passage. The line runs, on this account, from Tissot (1859/1881) through Grötzsch's 1930s work on distortion of non-conformal schlicht mappings, through Lavrentieff, to Ahlfors and Teichmüller's foundational 1930s–1940s development of quasiconformal mapping theory proper (Papadopoulos, 2017; Miyachi et al., 2025). This is, in other words, a case in which a specifically cartographic tool — built to answer a specifically cartographic question, how much does this particular projection of this particular country distort shape — turns out to sit at the root of a branch of twentieth-century complex analysis that has, on its own, nothing to do with maps.

## 30.4 From ellipse to bi-Lipschitz constant

The connection that matters most for the rest of this volume is not the genealogy itself but its most recent formalisation. Miyachi et al. (2025), following a reformulation due to Milnor (1969), replace the indicatrix's qualitative picture (an ellipse, read visually for its eccentricity and orientation) with an explicit pointwise quantity. For an open subset $\Omega$ of the sphere and a $C^1$-embedding $f : \Omega \to \mathbb{E}^2$ (a geographic map, in the ordinary sense), define at a point $x \in \Omega$

$$M(x) := \sup_{v} \frac{|df(v)|}{|v|}, \qquad m(x) := \inf_{v} \frac{|df(v)|}{|v|},$$

the supremum and infimum being taken over non-zero tangent vectors $v$ at $x$. These two numbers are precisely the lengths of the major and minor semi-axes of Tissot's own indicatrix at $x$, rescaled by the radius of the source circle; $M(x)$ and $m(x)$ coincide exactly when the indicatrix degenerates to a circle, i.e. exactly at the points where $f$ is conformal. Miyachi et al. (2025) then define the *infinitesimal bi-Lipschitz constant* of $f$ at $x$ as

$$\sigma(x) := \max\{\, M(x),\ m(x)^{-1} \,\},$$

and, taking the supremum and infimum of $\sigma$ over all of $\Omega$,

$$L(f) := \sup_{x \in \Omega} \sigma(x), \qquad \ell(f) := \inf_{x \in \Omega} \sigma(x), \qquad \nu(f) := \log\left(\frac{L(f)}{\ell(f)}\right),$$

the last quantity, $\nu(f)$, being what they call the *metrical distortion* of the map $f$ over $\Omega$ (Miyachi et al., 2025).

**Key Result**

The Tissot indicatrix and the bi-Lipschitz apparatus of Miyachi et al. (2025) are the same mathematical content in two different registers. $M(x)$ and $m(x)$ are the indicatrix's own axis lengths; $\sigma(x)$ collapses the ellipse's two-number shape description into the single worst-case stretch-or-compress factor at $x$; and $\nu(f)$ globalises that single number into a worst-case statement over the whole region — the ratio, on a log scale, between the map's most locally distorted point and its least distorted one. Nothing here is specific to DGGS or to any tessellation; it is a restatement, in the vocabulary of bi-Lipschitz embeddings, of exactly the same continuous-map distortion problem that Monographs 1–4 of this series approached via information-theoretic and metric-embedding routes, and that Tissot approached via an ellipse drawn by hand in 1859.

This reformulation is not merely notational tidiness. Miyachi et al. (2025) apply $\nu(f)$ (among other distortion measures) to compare the classical Marinus–Ptolemy and Delisle–Euler conical map families, and report that the eighteenth-century Delisle–Euler construction — arrived at by Delisle heuristically and only later given a mathematical treatment by Euler — comes out ahead of the alternatives they test on both the metrical and the quasiconformal criteria (Miyachi et al., 2025). The pattern they draw attention to, a craft solution predating by decades or centuries the mathematical proof of its own optimality, is the same pattern Miyachi et al. (2025) note for the azimuthal equidistant projection: in use for centuries beforehand, its optimality with respect to metrical distortion among conformal maps was proved only in 1969, by Milnor (1969), in the paper that itself introduced the $M(x)/m(x)$ formalism reused above (Miyachi et al., 2025).

## 30.5 What this reformulation is not

**To Complete / Verify**

Two boundaries are worth stating explicitly rather than leaving implicit. First, $\sigma(x)$, $L(f)$, $\ell(f)$ and $\nu(f)$ as defined above are properties of a *continuous* embedding $f$ on an open subset of the sphere; nothing in Section 30.4 constructs a discrete analogue for a DGGS cell complex, and no such construction is attempted in this chapter. A natural question — does a cell-wise version of $\sigma$, evaluated over the finite set of cells of a DGGS at a given resolution rather than over all points of a continuous $\Omega$, recover anything resembling the IPQ and areal statistics already computed in Chapters 11 to 16 of this volume — is left open, and no claim about it is made here. Second, the historical claim in Section 30.3 is Teichmüller's own attribution as reported by Papadopoulos (2017) and Miyachi et al. (2025); this volume does not independently verify the attribution against Teichmüller's original 1940 text, which was not consulted directly in this pass.

## 30.6 Closing the loop back to Chapter 1

Chapter 1 opened this volume by setting the continuous distortion problem, measured by the indicatrix, against the discrete distortion problem, measured cell by cell. The material above does not close that gap — Section 30.5 says explicitly that it does not — but it does sharpen one side of it. The indicatrix is not merely a picture; it is a specific, nameable pair of numbers at each point, and those numbers admit exactly the sup/inf, worst-case-ratio treatment that this volume's discrete chapters apply to cell area and shape statistics. The two halves of the continuous/discrete split that structures this book turn out, at the level of the mathematics used to state them, to be closer to each other than Chapter 1's original framing suggested.

Chapter 31

# The Indicatrix, Computed: Local Distortion in S2 and rHEALPix

## 31.1 Scope: from the tool revisited to the tool applied

Chapter 30 revisited the Tissot indicatrix as a mathematical object – what it is, who built it, what happened to the general theory behind it – without computing a single indicatrix for any of this volume's own grids. Chapter 19, in turn, compared S2's, rHEALPix's, and the ISEA family's continuous sphere-to-face maps by transcribing figures from each library's own documentation (a 5.2-to-1 area ratio for a naive linear cube projection against 2.08-to-1 for S2's quadratic correction, per Google (2024)) rather than by measuring anything directly. This chapter closes that gap for two of the five grids: it computes an actual Tissot indicatrix – the local scale factors $a, b$ and the maximum angular deformation $\omega$, via the general finite-difference method of Snyder (1987) – directly from S2's and rHEALPix's own continuous forward transforms, at a sequence of real points, rather than from a transcribed summary figure.

## 31.2 Method

**Methodological Note**

**What was done, precisely, and its limits.** For rHEALPix, the library's own `RHEALPixDGGS.rhealpix(lon, lat, inverse=False)` (the same continuous forward map already used, in its inverse direction, in Section 99.2) was evaluated on the unit sphere (`rhealpixdggs.dggs.UNIT_SPHERE`, $N_side = 3$, both square parameters 0) at four points bracketing each target $(\lambda, \phi)$, spaced $\pm 10^{-4}$ degrees in longitude and latitude respectively, to build a numerical Jacobian $(\partial x/\partial\phi, \partial y/\partial\phi, \partial x/\partial\lambda, \partial y/\partial\lambda)$ by central differences. For S2, the composite forward map $(\lambda, \phi) \mapsto (s, t)$ was built from `s2sphere`'s own `xyz_to_face_uv` (the linear, gnomonic cube-face coordinate) followed by `CellId.uv_to_st` under `QUADRATIC_PROJECTION` (the exact formula already quoted, from the same source, in Chapter 19) — i.e. the map from a uniform grid-index square to the sphere, inverted, since $(s, t)$ rather than the linear $(u, v)$ is what S2's actual cell grid is uniform in. The same central-difference Jacobian was then built from $(s, t)$ in place of $(x, y)$. Both Jacobians feed the same general formulas: $h = \|\partial(x, y)/\partial\phi\|$, $k = \|\partial(x, y)/\partial\lambda\|/\cos\phi$, $\sin\theta'$ from the Jacobian determinant, and $a, b$ from $h^2 + k^2$ and $hk\sin\theta'$ following Snyder (1987)'s general (non-conformal) case; $\omega = 2\arcsin\left(\frac{a-b}{a+b}\right)$.

One genuine error is disclosed because it was caught only after producing a first, wrong result: the first rHEALPix transect was centered on $\lambda = 0°$, which `RHEALPixDGGS.cell_from_point` confirms sits exactly on the boundary between two equatorial faces, not inside one. Because the equatorial belt's cylindrical-equal-area map turns out to be a function of latitude alone (checked directly: identical indicatrix val-

ues were obtained at $\lambda = 0°$ and $\lambda = 45°$ for the same latitude), this particular seam happened not to corrupt the equatorial-belt figures below – but the polar-cap figures, where the map is genuinely two-dimensional, do differ with longitude, and $\lambda = 45°$ (confirmed, by direct query, to sit inside a single face and a single polar-cap facet, away from every $0°/90°/180°/270°$ seam) is used throughout instead of the uncorrected $0°$. Every value below was also checked for step-size stability at $\varepsilon \in \{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ degrees; all agree to at least five significant figures at every point tested, including the two polar-cap points nearest a possible fold ($\phi = 45°, 85°$), evidence that no remaining seam was straddled. Full code: `code/tissot_s2_rhealpix.py`; full output: `data/tissot_results.json`, `data/tissot_convergence.json`.
This chapter does not attempt H3, DGGRID, or OpenEAGGR: none of the three installed libraries used elsewhere in this volume exposes a public, continuous (non-cell-quantized) forward transform analogous to rHEALPix's `rhealpix()` or S2's face/quadratic machinery, and none is reconstructed here from the ISEA formulas cited but not re-derived in Chapter 19. This is a real scope limit, not a finding.

## 31.3 rHEALPix: exactly equal-area everywhere, unevenly shaped

Table 31.1: rHEALPix Tissot indicatrix, $\lambda = 45°$, unit sphere, $N_side = 3$. Source: `code/tissot_s2_rhealpix.py`, `data/tissot_results.json`.

| $\phi$ (lat, °) | $a$ | $b$ | $\omega$ (°) | region |
|---|---|---|---|---|
| 0 | 1.17810 | 1.00000 | 9.380 | quad (equatorial belt) |
| 5 | 1.17361 | 1.00382 | 8.945 | quad |
| 10 | 1.16020 | 1.01543 | 7.631 | quad |
| 15 | 1.13795 | 1.03528 | 5.416 | quad |
| 20 | 1.10705 | 1.06418 | 2.263 | quad |
| 25 | 1.10338 | 1.06772 | 1.882 | quad |
| 30 | 1.15470 | 1.02026 | 7.088 | quad |
| 35 | 1.22077 | 0.96504 | 13.438 | quad |
| 40 | 1.30541 | 0.90247 | 21.031 | quad, near cap boundary |
| 45 | 1.32565 | 0.88869 | 22.762 | polar cap |
| 55 | 1.28418 | 0.91739 | 19.181 | polar cap |
| 65 | 1.25448 | 0.93911 | 16.532 | polar cap |
| 75 | 1.23531 | 0.95368 | 14.784 | polar cap |
| 85 | 1.22591 | 0.96100 | 13.915 | polar cap |
| 89 | 1.22479 | 0.96188 | 13.811 | polar cap, near-singular |

**Key Result**

The area scale factor $ab \sin\theta'$ – not tabulated above because it is, to seven significant figures, the same number at all fifteen points checked, equatorial and polar alike: $1.1780972\ldots$, varying only in the seventh digit across a full range of latitude from the equator to $0.011°$ from the pole. This is an independent, continuous-map confirmation of the exact equal-area property already established by discrete cell-counting throughout Part III's rHEALPix chapters: it holds not just for the finitely many realized cells at any one resolution, but for the underlying continuous map at every point tested, which discrete cell counting alone cannot show. Shape, by contrast, is not constant: $\omega$ falls from $9.38°$ at the equator to a minimum of $1.88°$ near $\phi = 25°$, then rises past $21°$ at the belt boundary, continues rising to a

maximum of $22.76°$ just inside the polar cap ($\phi = 45°$), and then *falls* again, monotonically, to $13.81°$ near the pole itself. Equal-area and low angular distortion are bought at different places on this grid: the point of least shape distortion in the belt ($\phi \approx 25°$) is not the point of greatest area uniformity (uniform everywhere, by construction) and is far from either the equator or the pole.

## 31.4 S2: distortion of the actual cell grid, not the raw cube face

Table 31.2: S2 Tissot indicatrix for the map from grid-index space $(s, t)$ to the sphere, face 0, diagonal transect $\lambda = \phi =$ step. Source: `code/tissot_s2_rhealpix.py`, `data/tissot_results.json`.

| step (°, from face center) | $a$ | $b$ | $\omega$ (°) | area scale |
|---|---|---|---|---|
| 0 | 0.75000 | 0.75000 | 0.000 | 0.56250 |
| 5 | 0.67763 | 0.67247 | 0.437 | 0.45569 |
| 10 | 0.64401 | 0.62454 | 1.758 | 0.40221 |
| 15 | 0.63904 | 0.59613 | 3.982 | 0.38095 |
| 20 | 0.65961 | 0.58240 | 7.127 | 0.38416 |
| 25 | 0.70705 | 0.58115 | 11.217 | 0.41090 |
| 30 | 0.78692 | 0.59184 | 16.268 | 0.46573 |
| 35 | 0.91020 | 0.61529 | 22.294 | 0.56004 |
| 40 | 0.96567 | 0.62458 | 24.771 | 0.60314 |
| 44 | 0.86928 | 0.60398 | 20.748 | 0.52503 |

**Key Result**

At the face center (step $= 0$, the point $\lambda = \phi = 0°$, confirmed by construction to be the exact center of S2 face 0), the indicatrix is a circle ($a = b, \omega = 0°$): the quadratic transform is locally isotropic at the one point it is explicitly designed to leave essentially unchanged (Google (2024)'s own $u = v = 0$ reference point). Moving along the diagonal toward the face corner, $\omega$ grows to $24.77°$ at step $= 40°$ — then *drops* to $20.75°$ at step $= 44°$, a full $4°$ closer to the corner. The area scale factor is similarly non-monotonic, falling from $0.5625$ at the center to a minimum of $0.3810$ around step $= 15°$, rising back past the center's own value to $0.6031$ at step $= 40°$, then falling again to $0.5250$ at step $= 44°$. Both reversals happen in the same last interval, immediately before the exact cube corner (step $= 45°$, where three faces meet and the map is singular); this chapter did not investigate why, and reports the reversal as observed rather than explained. This local, continuous-map quantity is not the same measure as Chapter 19's transcribed $2.08$-to-1 area ratio (a whole-face, realized-cell figure) and the two should not be read as confirming or contradicting each other; the two area-scale extremes measured here along one diagonal, $0.3810$ and $0.6031$, give a ratio of $1.58$, a different quantity computed by a different method along a single transect, not a whole-face extremum search.

## 31.5 Synthesis: two different shapes of the same problem

Both grids show angular distortion that is far from monotone in distance from wherever each construction's best-behaved point sits (S2's face center; rHEALPix's belt point near $\phi \approx 25°$), but the two patterns are structurally different from each other. S2's $\omega$ rises smoothly and

(mostly) monotonically away from its single isotropic center, consistent with a construction whose one explicit design goal is to be good at the center and to spend its one free correction (the quadratic warp) making that degradation as slow as practical. rHEALPix's $\omega$ is non-monotonic along its own transect, with an *interior minimum* away from both the equator and the pole, and never reaches zero anywhere sampled: consistent with a construction whose one explicit design goal is exact area preservation everywhere, with shape left to fall out however it falls out, including apparently accidental local optima with no obvious minimum. Neither grid is shown here to be more or less distorted than the other in any single number – they are optimizing for genuinely different things, which is the same conclusion Chapter 19 reached from the two constructions' stated design principles alone, now given an actual angular-distortion field rather than a single transcribed ratio to back it.

**To Complete / Verify**

This chapter checked one transect per grid (rHEALPix along its belt and into its polar cap at a single longitude; S2 along one face diagonal), not the full two-dimensional distortion field either grid's construction actually has. Whether rHEALPix's interior $\omega$ minimum near $\phi \approx 25°$ is a genuine extremum of the full two-dimensional field or an artifact of the single longitude sampled is not established here. **Resolved in Chapter 32:** the S2 reversal in the last $5°$ before the nominal face corner is not a property of the composite gnomonic-then-quadratic map at all – it is a face-boundary-crossing artifact of the diagonal transect, which silently leaves face 0 for a neighboring face at $\theta_c = \arcsin \frac{\sqrt{5}-1}{2} \approx 38.173°$, almost $7°$ before the corner; face 0's own map, derived there in closed form, is monotonically worsening (past one genuine interior area-scale minimum near $\theta \approx 16.8°$) all the way to where it actually stops applying. Extending this chapter's method to the ISEA family (H3, DGGRID, OpenEAGGR) would require either locating and verifying a primary source for Snyder's icosahedral equal-area formula in closed form, or exposing a continuous (non-cell-quantized) forward transform from one of the three libraries already built elsewhere in this volume, neither of which was attempted here.

Chapter 32

# The Corner Reversal, Derived

## 32.1 Scope: closing the open todobox item

Chapter 31 left one question explicitly open: along S2's face-diagonal transect, $\omega$ and the area scale factor grow away from the face center and then, in the last $5°$ before the nominal cube corner, *reverse* – $\omega$ falls from $24.771°$ at step $= 40°$ to $20.748°$ at step $= 44°$, and the area scale falls from $0.60314$ to $0.52503$ over the same interval. That chapter's own todobox called for "a derivation from the composite gnomonic-then-quadratic map's own Jacobian, rather than a further numerical scan." This chapter supplies that derivation, in closed form, and the answer it gives is not the one the open question presupposed: **there is no reversal in face 0's own map**. The reversal in Chapter 31's table is an artifact of the sampling method, not a property of the composite map near a corner.

## 32.2 Method: the composite map in closed form

**Methodological Note**

**What was done, precisely, and its limits.** Face 0 of the S2 cube (the face containing the $+x$ axis) is reached from geographic coordinates by the standard central/gnomonic tangent-plane construction: a point at $(\lambda, \phi)$ has unit vector $p = (\cos\phi\cos\lambda,\ \cos\phi\sin\lambda,\ \sin\phi)$, and face 0's raw (linear) face coordinates are $u = y/x = \tan\lambda$ and $v = z/x = \tan\phi\sec\lambda$ – independent of any library call, this is just the definition of gnomonic projection from the sphere's center onto the plane $x = 1$. S2's quadratic correction (already quoted from Google (2024) in Chapter 19 and used numerically, via `s2sphere.CellId.uv_to_st`, throughout Chapter 31) is then applied coordinatewise, $S = S(u)$, $T = T(v)$ with $S(w) = \frac{1}{2}\sqrt{1+3w}$ for $w \geq 0$. Because $S$ depends on $u$ alone and $T$ on $v$ alone, and $u, v$ are elementary functions of $(\lambda, \phi)$, the full $2 \times 2$ Jacobian $\partial(S,T)/\partial(\lambda,\phi)$ can be written down by direct differentiation (verified symbolically with `sympy`, not by hand algebra alone) rather than estimated by finite differences:

$$\frac{\partial S}{\partial \lambda} = \frac{3}{4\cos^2\lambda\sqrt{3\tan\lambda+1}}, \qquad \frac{\partial S}{\partial \phi} = 0, \tag{32.1}$$

$$\frac{\partial T}{\partial \lambda} = \frac{3\sin\lambda\tan\phi}{4\cos^2\lambda\sqrt{1+3\tan\phi\sec\lambda}}, \qquad \frac{\partial T}{\partial \phi} = \frac{3}{4\cos\lambda\cos^2\phi\sqrt{1+3\tan\phi\sec\lambda}}. \tag{32.2}$$

On the diagonal transect $\lambda = \phi = \theta$ used throughout Chapter 31, these feed the same general Tissot formulas already used there ($h, k$ from the Jacobian rows, $\sin\theta'$ from its determinant, $a, b$ and $\omega$ following Snyder (1987)), now evaluated from a closed-form expression rather than a central difference. This closed-form Jacobian is valid *only while the point is actually on face 0* – i.e. while $x = \cos^2\theta$ remains the largest-magnitude component of $p$. Two computations were run and compared, both in `code/s2_corner_reversal_explained.py`: (i) the closed-form face-0 Jacobian evaluated

at every $\theta$ up to $44°$, including values of $\theta$ at which the point has actually left face 0 (a deliberate, clearly labeled out-of-domain extrapolation, used only to show what face 0's own map would have done had the transect stayed on it); and (ii) the same auto-face-selecting method Chapter 31 itself used (`s2sphere.xyz_to_face_uv`, which silently reassigns each point to whichever face has the largest $|component|$), reproducing that chapter's original table exactly as a check. Full output: `data/s2_corner_reversal_explained.json`.

## 32.3 The face-crossing angle, in closed form

On the diagonal, $p = (\cos^2\theta,\ \cos\theta\sin\theta,\ \sin\theta)$. Face 0 requires $x \geq |z|$, i.e. $\cos^2\theta \geq \sin\theta$, i.e. $\sin^2\theta + \sin\theta - 1 \leq 0$. The boundary is the positive root of $\sin^2\theta_c + \sin\theta_c - 1 = 0$:

$$\sin\theta_c = \frac{\sqrt{5}-1}{2} \approx 0.618034, \qquad \theta_c = \arcsin\frac{\sqrt{5}-1}{2} \approx 38.173°. \tag{32.3}$$

The constant is the reciprocal of the golden ratio, $\sin\theta_c = 1/\varphi$ – a coincidence of this particular symmetric diagonal parametrization, not a claim about any deeper structure in S2's design. The important fact is simply that $\theta_c \approx 38.17°$ is *not* the cube corner $(45°)$: along the diagonal, the point leaves face 0 for a neighboring face (numerically confirmed, via direct `xyz_to_face_uv` calls, to be face 2) almost 7° before the corner is reached. Chapter 31's transect, sampled at step $\in \{35, 40, 44\}°$, straddles this crossing without having checked for it: $35°$ and $38°$ are still on face 0; $39°$ through $44°$ are already on face 2.

Table 32.1: Face-0 closed-form Jacobian (columns 2–3) versus the auto-face-selected values Chapter 31 actually tabulated (columns 4–5, face index in column 6), on the same diagonal transect. Source: `code/s2_corner_reversal_explained.py`, `data/s2_corner_reversal_explained.json`.

| step (°) | $\omega_{\text{face 0, forced}}$ | $\text{area}_{\text{face 0, forced}}$ | $\omega_{\text{auto}}$ | $\text{area}_{\text{auto}}$ | face |
|---|---|---|---|---|---|
| 35 | 22.294° | 0.56004 | 22.294° | 0.56004 | 0 |
| 38 | 26.379° | 0.64442 | 26.379° | 0.64442 | 0 |
| 38.17 | 26.621° | 0.65002 | 26.621° | 0.65002 | 0 |
| 39 | 27.819° | 0.67879 | 25.785° | 0.62781 | 2 |
| 40 | 29.298° | 0.71691 | 24.771° | 0.60314 | 2 |
| 42 | 32.374° | 0.80629 | 22.750° | 0.56038 | 2 |
| 44 | 35.604° | 0.91723 | 20.748° | 0.52503 | 2 |

**Key Result**

Up to and including $\theta_c \approx 38.173°$ the two columns agree to machine precision, as they must (both are evaluating face 0's own map, one by finite difference and one in closed form). Past $\theta_c$ they diverge immediately and sharply, in opposite directions: the honest continuation of face 0's own formula keeps *rising*, smoothly, all the way to step $= 44°$ ($\omega$ from $26.6°$ to $35.6°$, area scale from $0.650$ to $0.917$) – consistent with the ordinary intuition that a gnomonic-based cube map should get worse, not better, closer to a corner. The auto-face-selected values that Chapter 31 actually reported are, from step $= 39°$ on, not face 0's Jacobian at all: they are face 2's Jacobian, evaluated near *face 2's own* boundary (at face-2 coordinates $u \approx -0.96$ to $-0.74$, $v \approx -0.78$ to $-0.72$ – confirmed directly from `xyz_to_face_uv`'s own return values, not inferred), a region of face 2's patch with no reason to inherit face 0's corner-approach behavior. The reported "reversal" is therefore fully

explained: it is not a mathematical feature of the composite gnomonic-then-quadratic map near a cube corner, but a sampling artifact of a diagonal transect that crosses an unflagged face boundary $6.8°$ before reaching the corner it was nominally approaching.

## 32.4 A genuine feature, for contrast: the interior area-scale minimum

Not every non-monotonicity in Chapter 31's S2 table is an artifact. The area scale factor also has an interior *minimum* near step $= 15°$ ($0.3810$, against $0.4557$ at $10°$ and $0.3842$ at $20°$), well inside face 0 and far from the $38.17°$ crossing. Differentiating the closed-form area scale $d(\theta) = \big(\partial S/\partial\lambda\big)\big(\partial T/\partial\phi\big)/\cos\theta$ (both factors positive on the diagonal, so this already equals the Jacobian determinant divided by $\cos\phi$) and solving $d'(\theta) = 0$ numerically for the closed-form expression gives a single root on $(0, \theta_c)$ at

$$\theta^* \approx 16.794°, \qquad d(\theta^*) \approx 0.37946, \tag{32.4}$$

matching the tabulated minimum to three significant figures at the nearest sampled point ($15°$, $0.3810$) once the $5°$ sampling grid is accounted for. Unlike the corner reversal, this minimum survives the closed-form check: it is a genuine feature of face 0's own composite map, arising from the ordinary calculus fact that $\partial S/\partial\lambda$ is a product of a factor that grows with $\theta$ (from the raw gnomonic $\sec^2\lambda$) and a factor that shrinks with $\theta$ (the quadratic correction's own derivative, which decreases as its argument $u$ grows), and analogously for $\partial T/\partial\phi$; two such products, added and rescaled by $1/\cos\theta$, need not be monotonic even though every elementary factor in them is. $\omega$, by contrast, has no interior extremum on $(0, \theta_c)$ at all: its derivative, checked in closed form at nine points from $1°$ to $38°$, is strictly positive and strictly increasing throughout (Table 32.2), consistent with $\omega$ approaching, not receding from, its corner-adjacent maximum right up to the face boundary.

Table 32.2: $\omega$ and $d\omega/d\theta$ from the closed-form Jacobian, face 0 only. No sign change: $\omega$ is strictly increasing throughout face 0's own domain. Source: `code/s2_corner_reversal_explained.py`.

| $\theta$ (°) | $\omega$ (°) | $d\omega/d\theta$ (per rad) |
|---|---|---|
| 1 | 0.017 | 0.035 |
| 10 | 1.758 | 0.354 |
| 20 | 7.127 | 0.723 |
| 30 | 16.268 | 1.107 |
| 35 | 22.294 | 1.303 |
| 38 | 26.379 | 1.421 |

## 32.5 What this changes in Chapter 31

Chapter 31's own keyresult for S2 described the reversal as observed but uninvestigated, and its todobox listed the derivation attempted here as unattempted. Both statements should now be read as superseded by this chapter, not as still-open: the mechanism is face-crossing, not corner geometry, and face 0's own map is well-behaved (monotonically worsening in both $\omega$ and area scale, past a single genuine interior area-scale minimum near $\theta \approx 16.8°$) all the way to where it actually stops applying. Chapter 31's description of step $= 45°$ as the map's singular point remains correct as a statement about the cube corner itself, but is now known to

be a description of a point the diagonal transect, as originally sampled, never actually reaches while still on face 0.

**To Complete / Verify**

This chapter checked only the single diagonal transect $\lambda = \phi = \theta$, matching Chapter 31's own scope; it does not establish where the face-0/face-2 boundary falls in general (away from the diagonal) nor whether analogous crossings explain any non-monotonicity elsewhere in this volume's S2 tables. The golden-ratio coincidence in $\sin\theta_c$ is noted but not pursued mathematically; it does not obviously generalize to non-diagonal transects and no claim is made that it should.

Chapter 33

# The Ellipsoid, Not Assumed Away: rHEALPix's Indicatrix on WGS84

## 33.1 Scope: closing a gap left open twice

Chapter 31 computed a real Tissot indicatrix for rHEALPix from the library's own continuous forward transform, but evaluated it on `rhealpixdggs.dggs.UNIT_SPHERE`, not on the WGS84 ellipsoid this volume otherwise uses throughout (Chapter 27). Chapter 85's own todobox named the gap explicitly and left it open through two subsequent update notes that closed its other open items: "the unit-sphere-versus-WGS84 gap disclosed above for rHEALPix's Jacobian remains open ...a genuinely ellipsoidal rHEALPix Tissot field ...has not been built anywhere in this volume." This chapter builds it.

**Methodological Note**

**What was done, precisely, including a false start.** The library call itself changes by exactly one argument: `RHEALPixDGGS(ellipsoid=WGS84_ELLIPSOID, ...)` in place of `RHEALPixDGGS(ellipsoid=UNIT_SPHERE, ...)`, at the identical transect (longitude $45°$, the same latitude sequence, and the same $\pm 10^{-4}$ degree central-difference step already used and step-size-validated in Chapter 31). What does *not* carry over unchanged is the Tissot formula itself. A first attempt reused Chapter 31's spherical formula verbatim, normalizing both the meridional and the parallel scale factor by the single authalic radius $R_A$. This is correct for a genuine sphere, where one radius serves both directions, but not for an ellipsoid, whose meridian and parallel have different radii of curvature. The diagnostic that caught it: rHEALPix is exactly equal-area by construction, so its area-scale factor $h\,k\,\sin\theta'$ must come out constant across latitude; the first attempt's own area-scale factor did not (it ranged from $1.1728$ at the equator to $1.1887$ near the pole), which is impossible for a genuinely equal-area map and was the signal that the normalization, not rHEALPix, was wrong.

The fix uses Snyder (1987)'s own ellipsoidal Tissot formulas rather than the spherical special case: the meridional radius of curvature $M(\phi) = a(1-e^2)/(1-e^2\sin^2\phi)^{3/2}$ normalizes the $\phi$-direction scale factor $h$, and the prime-vertical radius of curvature $N(\phi) = a/\sqrt{1-e^2\sin^2\phi}$, combined with $\cos\phi$, normalizes the $\lambda$-direction scale factor $k$ — $a$, $e$ taken from `WGS84_ELLIPSOID`, the same object whose authalic radius $R_A = 6{,}371{,}007.18$ m was already cross-checked against Chapter 27's cited EPSG figure in Chapter 85. With this correction, the raw Jacobian determinant of the ellipsoidal `rhealpix()` output, divided by the true ellipsoidal area element $M(\phi)N(\phi)\cos\phi$, reproduces the unit-sphere area-scale constant $1.1780972451\ldots$ to ten significant figures at every one of the fifteen latitudes checked — the correction is confirmed by the same equal-area identity that first exposed the error. Step-size stability was rechecked at $\phi \in \{0°, 25°, 45°, 89°\}$ for $\varepsilon \in \{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ degrees; all agree to at least five significant figures. Full code: `code/rhealpix_ellipsoidal_tissot.py`;

full output, including the disclosed false-start numbers alongside the corrected ones: `data/rhealpix_ellipsoidal_tissot_results.json`.
This chapter does not attempt S2: `s2sphere`, the library used throughout this volume for S2, implements the unit-sphere cube-face construction only and exposes no ellipsoidal variant, so there is no analogous "one-argument change" available for it. Extending S2 to a genuinely ellipsoidal Tissot field would require reconstructing its face/quadratic machinery over an ellipsoidal auxiliary-sphere substitution from scratch, not attempted here.

## 33.2 The equatorial belt: distortion grows, in both directions, away from the equator

Table 33.1: rHEALPix Tissot indicatrix, $\lambda = 45°$, $N_side = 3$: unit sphere versus WGS84 ellipsoid (corrected formula). Source: `code/rhealpix_ellipsoidal_tissot.py`, `data/rhealpix_ellipsoidal_tissot_results.json`.

| $\phi$ (lat, °) | $\omega_{\text{unit}}$ (°) | $\omega_{\text{WGS84}}$ (°) | $\Delta\omega$ (°) | region |
|---|---|---|---|---|
| 0 | 9.380 | 9.508 | +0.128 | quad (equatorial belt) |
| 5 | 8.945 | 9.076 | +0.131 | quad |
| 10 | 7.631 | 7.770 | +0.139 | quad |
| 15 | 5.416 | 5.570 | +0.154 | quad |
| 20 | 2.263 | 2.436 | +0.173 | quad |
| 25 | 1.882 | 1.686 | −0.197 | quad |
| 30 | 7.088 | 6.864 | −0.224 | quad |
| 35 | 13.438 | 13.185 | −0.253 | quad |
| 40 | 21.031 | 20.748 | −0.282 | quad, near cap boundary |

Table 33.2: rHEALPix Tissot indicatrix, polar cap, $\lambda = 45°$: unit sphere versus WGS84 ellipsoid (corrected formula). Same source as Table 33.1.

| $\phi$ (lat, °) | $\omega_{\text{unit}}$ (°) | $\omega_{\text{WGS84}}$ (°) | $\Delta\omega$ (°) | region |
|---|---|---|---|---|
| 45 | 22.762 | 22.751 | −0.011 | polar cap |
| 55 | 19.181 | 19.177 | −0.004 | polar cap |
| 65 | 16.532 | 16.531 | −0.001 | polar cap |
| 75 | 14.784 | 14.784 | −0.000 | polar cap |
| 85 | 13.915 | 13.915 | −0.000 | polar cap, near-singular |
| 89 | 13.811 | 13.811 | −0.000 | polar cap, near-singular |

**Key Result**

The ellipsoid substitution is not a rounding-order correction inside the equatorial belt: $\Delta\omega$ grows monotonically in magnitude from $+0.128°$ at the equator to $-0.282°$ at $\phi = 40°$, changing sign near $\phi \approx 22$–$25°$ — close to, but not exactly at, the belt's own interior $\omega$ minimum already identified in Chapter 31 near $\phi \approx 25°$. The sign change means the ellipsoid does not uniformly worsen or improve rHEALPix's angular distortion within the belt; it reshapes the transect, shifting the location and depth of the existing minimum rather than adding a fixed offset to it. Crossing into the polar cap at $\phi = 41.8°$, the effect collapses abruptly: from $-0.28°$ at the last belt point to $-0.01°$ immediately inside the cap, and further toward zero approaching the pole. This is consistent with the cap's own construction:

HEALPix's polar facets are built from a $\sigma(\phi) = \sqrt{3(1 - |\sin\phi|)}$ radial contraction that depends on $\phi$ only through $\sin\phi$, which the authalic substitution (Section 33.2's own $h(\phi)$, tabulated in Chapter 29) perturbs only slightly near the pole, where $\sin\phi$ itself is near-stationary.
Throughout both regions, the equal-area property survives the ellipsoid substitution exactly (area-scale constant to ten significant figures at every point checked) — as it must, since the authalic latitude is defined precisely so that it does (Section 28.2 and Chapter 29); what changes under the ellipsoid is only the *shape* term $\omega$, never the area term. rHEALPix's defining trade – exact area preservation, angular distortion left to fall out however it falls out – is unaffected by which reference surface the area is preserved *from*; only the specific numeric shape of that fallout shifts.

**To Complete / Verify**

This closes Chapter 85's remaining open item for rHEALPix specifically. Three narrower gaps remain, none claimed as closed here: (1) S2's own ellipsoidal field, blocked by `s2sphere`'s unit-sphere-only implementation, as disclosed above; (2) the transect is still the single longitude $\lambda = 45°$ that Chapter 31 used for the unit-sphere case, for the same face-boundary reason given there — whether $\Delta\omega$'s sign-change latitude shifts with longitude is not established; (3) H3, DGGRID, and OpenEAGGR remain outside this chapter's method for the same reason Chapter 31 gave: none of the three exposes a continuous, non-cell-quantized forward transform this method can differentiate.

Chapter 34

# Before Milnor: Euler, Gauss, and Chebyshev on Distortion

## 34.1 Scope: inevitability is not optimality

Chapter 30 traced a single lineage — Tissot's 1859/1881 indicatrix, its quasiconformal afterlife, and Milnor's 1969 optimality theorem for the azimuthal equidistant projection — and closed by calling that lineage "the same continuous-map distortion problem" this volume's earlier monographs approached by other routes. That chapter left one thing unexamined: Milnor's 1969 result is an *optimality* theorem, answering the question "which map, among a stated class, distorts least?" It presupposes, without proving, the prior and logically weaker claim that *every* map in that class distorts at least *something*. That prior claim — inevitability, not optimality — has its own history, older than Tissot's, and this chapter recovers it, together with one sibling optimality theorem, Chebyshev's, that predates Milnor's by over a century and answers a version of the same question for a different class of maps.

**Methodological Note**

This chapter introduces two new sources to Section 114.6: Osserman (2004), fetched and read in full in this pass, for Euler's 1775 and Gauss's 1827 results; and Pędzich (2005), likewise fetched and read in full, for Chebyshev's 1856 theorem and Grave's 1896 proof of it. Neither Euler's, Gauss's, nor Chebyshev's original texts are consulted directly; both are reported here exactly as Osserman (2004) and Pędzich (2005) respectively state them. No new computation is performed.

## 34.2 Euler, 1775: no exact-scale map of any part of a sphere

The weakest and oldest form of the inevitability claim is due to Leonhard Euler. Osserman (2004) states it as: *it is impossible to make an exact scale map of any part of a spherical surface* — meaning a map under which every pair of points is rescaled by exactly the same constant factor, the way an architect's floor plan rescales a building. Osserman (2004) attributes this to Euler's 1775 paper *On representations of a spherical surface on the plane* and notes that the claim was already an established empirical fact among cartographers well before Euler gave it a proof: Johann Lambert's 1772 treatise, the first general mathematical treatment of cartography, already distinguished size-preserving from shape-preserving maps and took for granted that no map could do both.

Euler's theorem, on Osserman (2004)'s account, is provable directly from the cylindrical-projection formulas this volume's other chapters take for granted: a cylindrical projection of the sphere maps a parallel of latitude $\phi$ with horizontal scale factor $s \sec \phi$ (Osserman, 2004, adapting Snyder's standard cylindrical-projection derivation), a factor that varies with $\phi$ and can therefore never be made a single constant across the whole map unless the map covers

only a single parallel. This is a narrow argument, restricted to cylindrical projections specifically, but Euler's own 1775 result was general, covering every possible mapping of any region of the sphere, not only the cylindrical family.

## 34.3 Gauss, 1827: the sphere is the rule, not the exception

Half a century later, Gauss generalised Euler's claim in a direction Euler could not have anticipated. Osserman (2004) summarises the result of Gauss's 1827 *Disquisitiones Generales Circa Superficies Curvas* — the paper containing what Gauss himself named the Theorema Egregium, the "remarkable theorem" — as follows: *if $S$ is any surface, with a very few exceptions, then there is no exact scale map of any region of $S$ onto the plane*. The exceptions are the *developable* surfaces — cylinders, cones, and a class called tangent developables — each obtainable by rolling a flat sheet of paper without stretching it. Every other surface, the sphere among them, inherits Euler's no-exact-scale-map conclusion not as a sphere-specific curiosity but as a consequence of a single intrinsic quantity Gauss's theorem shows to be preserved under any distance-and-angle-preserving deformation: Gaussian curvature.

**Key Result**

Euler's 1775 theorem and Gauss's 1827 theorem are not the same claim at two different levels of generality by accident; Gauss's theorem is specifically what upgrades Euler's sphere-only observation into a structural fact about curved surfaces as such. A plane has zero Gaussian curvature everywhere; a sphere of radius $R$ has constant positive curvature $1/R^2$; an exact-scale (isometric) map between two surfaces must, by the Theorema Egregium, preserve Gaussian curvature pointwise; since $1/R^2 \neq 0$, no region of a sphere can be mapped isometrically onto any region of a plane. Every DGGS this volume examines, and every classical projection Monographs 1–4 examined before it, operates inside the space of possibilities this one theorem closes off: none of them is exempt from Gauss's conclusion, and none of them could be.

## 34.4 Chebyshev, 1856: optimal, not merely inevitable, for conformal maps

Granting that no map is distortion-free, the next question — Tissot's question, per Section 30.2, and Milnor's after him — is which map, within a stated class, distorts *least*. Pędzich (2005) attributes to Pafnuty Chebyshev, in a talk to the Imperial Russian Geographic Society dated 1853 in Pędzich (2005)'s own running text (reported elsewhere in the literature as 1856; the discrepancy is not resolved here), a criterion and a theorem specifically for *conformal* map projections. Chebyshev measured distortion at a point by the natural logarithm of the local scale factor $m$, $z = \ln m$, and proposed as a criterion that the best conformal projection of a given region $F$ is the one minimising

$$\max_{(u,v)\in F,A} \ln m \; - \; \min_{(u,v)\in F,A} \ln m,$$

the difference between the largest and smallest values the log-scale attains anywhere in the region, over all points $(u, v)$ and directions (azimuths) $A$ (Pędzich, 2005). Chebyshev's 1856 theorem states the criterion's solution in a strikingly simple geometric form: a conformal projection of a given region has minimal-oscillation log-scale distortion *exactly when the scale factor is constant along the region's boundary* (Pędzich, 2005). Chebyshev's own proof was incomplete;

Pędzich (2005) reports that D. A. Grave supplied a full proof of the theorem for the conformal case in 1896, forty years after Chebyshev's original talk.

**To Complete / Verify**

Section 30.4 of Chapter 30 reports Miyachi et al. (2025)'s claim that Milnor's 1969 theorem establishes the azimuthal equidistant projection's optimality "among conformal maps." The azimuthal equidistant projection is not itself conformal (it preserves distance from the map's centre point, not angles), so read literally that phrase sits in some tension with the reading of Milnor's result given here, in which Chebyshev's theorem, not Milnor's, is specifically the one restricted to the conformal class. This volume has not fetched Milnor (1969) directly in either chapter and cannot resolve the tension from primary text; it is flagged here rather than silently harmonised, and both chapters should be revisited together once Milnor (1969) is consulted directly.

## 34.5 Two centuries, two kinds of theorem

Set side by side, the four results of this chapter and Chapter 30 fall into two logically distinct families, not one continuous escalation:

| Result | Year | Question answered |
|---|---|---|
| Euler | 1775 | Does an exact-scale map of a sphere region exist? (No.) |
| Gauss | 1827 | Does an exact-scale map of *any* curved surface exist? (No, with named exceptions.) |
| Chebyshev/Grave | 1856/1896 | Given inevitable distortion, which *conformal* map minimises it? |
| Tissot | 1859/1881 | How is distortion measured, pointwise, at all? |
| Milnor | 1969 | Which map, in a stated class, minimises Tissot's measure? |

Euler's and Gauss's results are *inevitability* theorems: they say a perfect map cannot exist, full stop, independent of any measure of how imperfect an imperfect map is. Chebyshev's and Milnor's are *optimality* theorems: each presupposes inevitability and asks, within a specific class of maps and a specific measure of distortion, which member of the class is least bad. Tissot's own contribution sits between the two families, supplying the pointwise measuring instrument (the indicatrix) that an optimality theorem needs before it can even state what it is optimising. This volume's title, *From Continuous to Discrete*, and its central claim that the same cost is paid whether distortion is managed continuously (Part I's global frame, Part II's regional case) or discretely (Part III's DGGS efficiency trade-offs), rests, at bottom, on the inevitability half of this table: Gauss's theorem is the reason there is a cost to manage at all, on any surface this volume has examined, continuous or tessellated.

Chapter 35

# From Indicatrix to Score: A Genealogy of the Cost of Distortion

## 35.1 Scope: what the earlier chapters left as a pointwise measure

Chapter 30 and Chapter 34 together established two things: that distortion is unavoidable on any map of any curved surface (Euler 1775, Gauss 1827), and that Tissot's 1859/1881 indicatrix supplies the pointwise instrument for measuring how much of it a given map carries at a given location, with Chebyshev's and Milnor's theorems each optimizing that instrument over a stated class of maps. What neither chapter addressed is a question this volume's own title raises directly: how did "the cost of distortion" turn from a pointwise geometric quantity, attached to a single infinitesimal circle, into the kind of single comparable number — a *score* — that lets two whole-world maps, or two whole-world discrete grids, be ranked against each other? This chapter traces that second lineage, from Tissot's own indicatrix through a twentieth-century political controversy that reframed distortion as a normative cost rather than a neutral geometric fact, to the explicit numerical error budgets in use today, which this volume's own DGGS trade-off chapters (Part III) inherit without, until now, naming their ancestry.

**Methodological Note**

This chapter introduces three sources to Section 114.6 fetched directly and read in full: Peters (1983)'s argument and Goldberg & Gott (2008)'s six-term error budget, both reported here via Gott, Goldberg & Vanderbei (2021), which restates each in enough technical and quotable detail to cite them without independently locating and fetching Peters (1983)'s 1983 book or Goldberg & Gott (2008)'s 2008 *Cartographica* article; and Gott, Goldberg & Vanderbei (2021) itself, fetched directly in full from arXiv. A fourth source, Kimerling et al. (1999), is cited only via its abstract and via its description in later secondary literature, since the original *Cartography and Geographic Information Science* article sits behind a publisher paywall not accessed in this pass; no numeric detail from it beyond what its own abstract states is used below. No new computation is performed in this chapter.

## 35.2 Tissot's indicatrix as a starting point, not an endpoint

Chapter 30 already gives Tissot's own construction in full; it is recalled here only for what it does *not* do. Tissot's indicatrix reports, at a single point, the shape and area an infinitesimal circle on the globe acquires once mapped to the plane — two numbers, semi-axes $a$ and $b$, from which local angular and areal distortion both follow. It says nothing, by itself, about how those two numbers should be combined into one, nor how a value computed at one point should be combined with values computed at every other point on the map into a single figure of merit for the whole projection. Every step in this chapter's genealogy is, in one way or another, an

answer to that combination problem: how to go from a field of local measurements to one global number, and, once a number exists, whether the number measures a technical quantity alone or something with normative or political weight attached.

## 35.3 Peters, 1983: distortion reframed as a normative cost

The clearest point at which "distortion" stopped being treated purely as a technical quantity and became explicitly a *cost* — something paid unequally by different parties, not merely present or absent — is Arno Peters's 1983 book *The New Cartography*. Per Gott, Goldberg & Vanderbei (2021)'s account, Peters presented an equal-area world projection (itself, as that source notes, a rediscovery of James Gall's 1885 projection) and argued that the Mercator projection, in wide institutional and classroom use at the time, was *culturally unfair to developing countries*, which it renders smaller on the page than their true surface area warrants relative to higher-latitude industrialized regions.

**Key Result**

Peters's argument is not, at bottom, a claim that the Mercator carries more distortion than some rival projection in any of the senses Chapter 30 or Chapter 34 formalize; conformal and equal-area projections are simply different, each inevitable in Gauss's sense and each optimal for a different criterion in Chebyshev's or Milnor's sense. Peters's contribution to this genealogy is instead the move of treating the *distribution* of a projection's areal error across the globe — who is shrunk and who is not — as itself a cost with a bearer, rather than treating total distortion as a single scalar to be minimized in the abstract. This is a different quantity from anything Tissot, Chebyshev, or Milnor measured, and it is the reason later attempts at a single numerical "distortion score," beginning with Section 35.5 below, still had to decide, as a normative rather than purely mathematical choice, how much weight to give area error relative to shape, distance, or other error terms.

Gott, Goldberg & Vanderbei (2021) also records that the Peters projection did not settle the argument it provoked: the same source notes plainly that the competing Winkel Tripel projection, which is not equal-area, "has not come in for much criticism" despite that property, suggesting that public and cartographic reception of a projection's distortion profile tracks something other than any single formal error measure on its own.

## 35.4 Toward a combinable measure: Laskowski's distortion spectrum

Between Peters's 1983 political intervention and the explicit numerical scoring systems of the following section, Gott, Goldberg & Vanderbei (2021) identifies a specific technical precedent for the idea of normalizing a projection's error terms against a reference projection before combining them: Laskowski's two-part 1997 *Cartographica* monograph, *Distortion-Spectrum Fundamentals* and *Distortion-Spectrum Applications*. Gott, Goldberg & Vanderbei (2021) cites this work as the methodological precedent its own normalization procedure follows, without itself restating Laskowski's fundamentals section in detail.

**To Complete / Verify**

This chapter reports only that Gott, Goldberg & Vanderbei (2021) names Laskowski's 1997 monograph as its normalization precedent; it does not independently fetch either 1997 *Cartographica* part and cannot report what distortion-spectrum visualization technique Laskowski introduced beyond that acknowledgement. A future revision should locate and read Laskowski (1997a, 1997b) directly before treating any technical claim beyond the bare precedent as established.

## 35.5 Goldberg and Gott, 2007/2008: six errors, one score

The genealogy's clearest technical culmination, on the evidence assembled here, is Goldberg & Gott (2008)'s error budget, as restated in full working detail by Gott, Goldberg & Vanderbei (2021). Where Tissot's indicatrix supplies two local numbers (isotropy and area, in Gott, Goldberg & Vanderbei's relabelling of Tissot's own quantities as $I$ and $A$), Goldberg & Gott (2008) add four more, extending the measurement from a single point to the geometry of whole geodesics and whole maps: *flexion* $F$, the bending on the map of what is a straight geodesic on the globe; *skewness* $S$, an asymmetric stretching along a geodesic that Gott, Goldberg & Vanderbei (2021) illustrates with a truck driving due north on a Mercator map, accelerating in speed (skewness) without turning its wheel (zero flexion, since its path stays straight); *distance error* $D$, the root-mean-square logarithmic discrepancy between map distance and true globe distance over random point pairs; and *boundary-cut error* $B$, a penalty proportional to the total angular length of the cuts a projection must introduce to lay the sphere flat.

**Key Result**

Goldberg & Gott (2008)'s decisive move, per Gott, Goldberg & Vanderbei (2021)'s account, is normalizing each of the six terms against the same reference projection (the equirectangular, following Gott, Goldberg & Vanderbei's report of Laskowski's precedent) and summing their squares into one number, $\Sigma\sigma^2$, with a perfect map (a literal globe) scoring $0$. This is the first point in the genealogy traced here at which "cost of distortion" becomes, without qualification, a single real number attached to an entire world map, additive across independent error types, and directly comparable between any two projections: the worked table Gott, Goldberg & Vanderbei (2021) reproduces gives the equirectangular a score of exactly $6.0$ by construction, the Mercator $8.308$, and the Winkel Tripel — at the time the best-scoring conventional flat projection by this metric — $4.563$.

The normalization choice is itself worth flagging as a design decision rather than a mathematical necessity, in the same spirit as Section 35.3's point about Peters: Gott, Goldberg & Vanderbei (2021) states outright that "the combination of the six error terms depends on the choice of normalizing projection," and that the equirectangular was chosen for this role only because Goldberg & Gott (2008) followed Laskowski's precedent in doing so, not because the equirectangular has any privileged mathematical status among the projections it is used to rank.

## 35.6 2021: driving the score down, and what stayed unchanged

Gott, Goldberg & Vanderbei (2021) treats the Goldberg & Gott (2008) score not as a description but as an explicit optimization target, in a sense closer to this volume's own DGGS effi-

ciency chapters (Part III) than to anything in Chapter 30's original, purely descriptive use of the indicatrix. Working within the conventional single-sided flat-map class, Gott, Goldberg & Vanderbei (2021) reports a marginal improvement over the Winkel Tripel (a "Gott-Wagener" variant scoring $4.497$ against the Winkel Tripel's $4.563$), then abandons the single-sided constraint altogether: a double-sided disk map, printed front and back like a phonograph record and assembled by folding, is reported to reach a score of $0.881$ for its best (azimuthal equidistant) variant, chiefly by eliminating the boundary-cut term $B$ outright (a double-sided disk has no cut) and sharply reducing the distance term $D$.

**Key Result**

Across a century and a half, the object being minimized has stayed the same in kind — some aggregate measure of how far a flat representation departs from the sphere it represents — while the space of objects allowed to compete for the minimum has widened at each step: Tissot measured single points; Goldberg & Gott (2008) measured whole conventional (single-sided, one-piece) flat projections; Gott, Goldberg & Vanderbei (2021) widened the competition to include a topologically distinct class of object (two-sided, foldable, zero-volume disks) that earlier distortion theory, built around a single continuous sheet, had no vocabulary to compare against a Mercator or a Winkel Tripel at all. This widening-of-the-competition pattern is the same one Chapter 20 traced for a different reason (S2's cube against H3's icosahedron as competing base solids for a discrete grid, not competing flat sheets), and it recurs a third time in this volume's own move from continuous projections (Part I's early chapters) to discretely tessellated DGGS cells (the remainder of Part I onward): each transition keeps "how much does fidelity to the sphere cost" as the question while changing what kind of object is allowed to answer it.

## 35.7 The same move made for discrete grids

Kimerling et al. (1999)'s abstract, independently confirmed across several secondary citations of the same 1999 *Cartography and Geographic Information Science* article, describes a global grid comparison study built around exactly this genealogy's combination problem, transposed from continuous map projections to the recursively tiled polyhedral DGGS this volume's own Part I treats at length: a set of evaluation criteria for comparing global tessellations, of which metrics for spheroidal surface area, cell compactness, and center-point spacing are reported, per the abstract, as "of particular importance."

**To Complete / Verify**

Only the abstract and secondary descriptions of Kimerling et al. (1999) have been consulted in this pass; the full text, and in particular whatever combination rule (if any) it uses to turn its several named criteria into a single ranking, has not been read directly and is not reported here. Given this volume's own recurring use, across Part III, of multiple separately reported distortion and efficiency metrics for DGGS without a single combined score, a future revision should read Kimerling et al. (1999) directly and determine whether it attempted (or explicitly declined) a Goldberg–Gott-style single-number synthesis for the discrete case, before citing it as either a precedent for, or a contrast with, this volume's own practice.

## 35.8 Closing: an unfinished genealogy, not a solved problem

Set end to end, the five points in this chapter — Tissot's pointwise indicatrix, Peters's normative reframing of areal error as an unequally distributed cost, Laskowski's normalization precedent, Goldberg & Gott (2008)'s six-term combined score, and Gott, Goldberg & Vanderbei (2021)'s widening of the competition to double-sided topology — do not converge on a single settled answer to "how much does distortion cost." They converge, instead, on an increasingly explicit statement of the *same three open choices* at each step: which error terms to include, how to normalize and weight them against one another, and what class of object is allowed to compete for the lowest score. Part III's own DGGS efficiency chapters make exactly these three choices again, for a discretely tessellated rather than continuously mapped Earth, generally reporting several distortion and timing metrics side by side rather than collapsing them into one Goldberg & Gott-style sum of squares. This chapter does not argue that either approach is mistaken; it records, for the first time in this volume, that the choice between them is not a contingent feature of any one chapter's method but the latest instance of a design decision every quantifier of "the cost of distortion" has faced since at least 1983, and arguably since Tissot's own 1859 choice of which two numbers, out of everything an indicatrix could report, to name.

Chapter 36

# Closing the Gap: The Indicatrix, Derived, for H3

## 36.1 Scope: the gap left open by Chapter 31

Chapter 31 computed a real Tissot indicatrix for S2 and rHEALPix directly from each library's own continuous sphere-to-face transform, and closed with an explicit refusal to attempt H3, DGGRID, or OpenEAGGR: none of the three exposes a public continuous forward map the way `rhealpixdggs.rhealpix()` and `s2sphere` do, and reconstructing one was, at that point, out of scope. This chapter removes that limit for H3 specifically, by deriving the continuous map H3 actually uses internally – an azimuthal gnomonic projection onto each of the twenty icosahedron faces – from H3's own public source, rather than by guessing at or approximating it.

**Methodological Note**

**What was done, and its sourcing.** H3's C library (Uber Technologies, 2026) (`https://github.com/uber/h3`, Apache License 2.0) exposes, in `src/h3lib/lib/faceijk.c`, the function `_vec3ToHex2d`: given a point on the sphere, it locates the nearest of twenty fixed icosahedron face-center points, forms the great-circle angular distance $c$ from that center, and sets the planar radial coordinate to $\rho = \tan(c)$ – an azimuthal gnomonic projection in polar form – before a further, purely azimuthal rotation (a fixed per-face reference angle, plus an additional constant $19.10661°$ rotation on "Class III" i.e. odd resolutions) and a purely radial, position-independent resolution-dependent rescaling ($\times 7^{-r/2}$, tracked in the source via repeated multiplication by $1/\sqrt{7}$). Twenty face-center unit vectors were transcribed directly from that source file for this chapter (array `faceCenterPoint`); no other part of the file's addressing machinery is used or reproduced. Because the azimuthal rotation and the radial rescaling are both *position-independent* on a given face – the same rotation angle and the same scale factor apply everywhere on that face – neither can alter the local angular deformation $\omega$ at a point: a fixed rotation only reorients the Tissot ellipse's axes, and a fixed positive radial scaling multiplies both semi-axes by the same constant, leaving their ratio (hence $\omega$) unchanged. This is argued here from the transformations' own structure, not assumed; it means the shape of H3's local distortion field is fully determined by the gnomonic radial law $\rho = \tan(c)$ alone, independent of face, orientation reference, and resolution. Full code: `code/tissot_h3_gnomonic.py`; full output: `data/tissot_h3_gnomonic_results.json`.

## 36.2 The closed-form solution

The azimuthal gnomonic projection, in the polar form H3 itself uses (radial coordinate $\rho = \tan c$, azimuth $\theta$ unchanged), has a classical closed-form Tissot solution, re-derived here di-

rectly from the map rather than transcribed from a table: the radial scale factor is $h = d\rho/dc = \sec^2 c$, and the transverse scale factor is $k = \rho/\sin c = \tan c/\sin c = \sec c$. Since $h \geq k \geq 1$ for all $c \in [0, \pi/2)$, the semi-axes are $a = \sec^2 c$, $b = \sec c$, giving

$$\omega = 2\arcsin\left(\frac{\sec^2 c - \sec c}{\sec^2 c + \sec c}\right) = 2\arcsin\left(\frac{1-\cos c}{1+\cos c}\right), \qquad \text{area scale} = hk = \sec^3 c.$$

Both the angular deformation and the area scale grow monotonically and unboundedly as $c \to \pi/2$ (the projection's well-known horizon singularity at $90°$ from the tangent point) – entirely consistent with, and here for the first time derived rather than merely cited for, the "documented $\pm 50\%$ H3 cell-size variation" figure this volume has referenced from library documentation elsewhere.

## 36.3 Verification: closed form against finite differences, face against face

Table 36.1: H3 gnomonic Tissot indicatrix, face 0, transect from the face center. $\omega$ from the closed form of Section 36.2; the finite-difference reconstruction (same central-difference method as Chapter 31, citing Snyder (1987)) agrees at every point to within $4 \times 10^{-9°}$ (last column). Source: `code/tissot_h3_gnomonic.py`.

| $c$ (°, from face center) | $\omega$ (°) | area scale | $\lvert\omega_{\text{closed}} - \omega_{\text{f.d.}}\rvert$ (°) |
|---|---|---|---|
| 0 | 0.000 | 1.0000 | 0 |
| 3 | 0.079 | 1.0041 | $1.8 \times 10^{-9}$ |
| 6 | 0.315 | 1.0166 | $6.2 \times 10^{-10}$ |
| 9 | 0.710 | 1.0379 | $3.3 \times 10^{-9}$ |
| 12 | 1.266 | 1.0685 | $3.3 \times 10^{-9}$ |
| 15 | 1.986 | 1.1096 | $2.9 \times 10^{-9}$ |
| 18 | 2.875 | 1.1625 | $9.4 \times 10^{-10}$ |
| 20.9 | 3.899 | 1.2265 | $3.5 \times 10^{-9}$ |

The residual agreement (at worst $3.5 \times 10^{-9}$ degrees, i.e. floating-point-level, not approximation-level) is itself the chapter's central check: it confirms that the closed-form gnomonic solution of Section 36.2 and an independent central-difference Jacobian of the same map agree to numerical precision, and were stable under step size $\varepsilon \in \{10^{-2}, \ldots, 10^{-5}\}°$ to at least six significant figures throughout, the same stability discipline Chapter 31 applied to S2 and rHEALPix.

A second, independent check addresses face-dependence directly: the same transect method was repeated from face 5's center (a different one of the twenty faces, verified via its own transcribed face-center vector to sit in a different part of the sphere from face 0) along a different bearing ($15°$ rather than due north). At matched great-circle distances $c = 6°, 12°, 18°$, the two faces agree on $\omega$ to five significant figures ($0.31473°$ vs. $0.31473°$; $1.26591°$ vs. $1.26591°$; $2.87490°$ vs. $2.87490°$), as the closed form of Section 36.2 – which depends on $c$ alone – predicts. This is not a face-specific result but a genuinely global one for the icosahedral gnomonic construction.

**Key Result**

The face's own circumradius (face-center to vertex angular distance), computed here from the transcribed face-center vectors of the five faces meeting at one icosahedron vertex ($\{0, 1, 2, 3, 4\}$, per H3's own face-adjacency structure) rather than looked up, is $37.377°$:

the gnomonic law of Section 36.2 therefore never comes close to its $90°$ singularity within a single face, and the modest area-scale range in Table 36.1 ($1.0$ at the center to $1.23$ at $20.9°$, i.e. well short of the true circumradius) is representative of the whole face, not a near-singular outlier.

## 36.4 An empirical cross-check against real H3 cells

The derivation above never calls `h3-py`; it works entirely from the transcribed face-center table and the standard gnomonic formula. As an independent check tying it back to the actual library, real H3 cells at resolution 6 were generated (`h3.latlng_to_cell`) at increasing target distances from face 0's center, and an empirical elongation proxy – the ratio of the farthest to the nearest of the six real boundary vertices (`h3.cell_to_boundary`) from the cell center, both measured as real great-circle distances – was compared against this chapter's predicted $h/k = \sec c$ ratio at the cell's actual center:

Table 36.2: Real H3 cells (res. 6) vs. predicted gnomonic elongation. Source: `code/tissot_h3_gnomonic.py`, `data/tissot_h3_gnomonic_results.json`.

| target $c$ (°) | actual cell-center $c$ (°) | empirical vertex-dist. ratio | predicted $h/k = \sec c$ |
|---:|---:|---:|---:|
| 0 | 0.000 | 1.0000 | 1.0000 |
| 5 | 5.010 | 1.0033 | 1.0038 |
| 10 | 10.001 | 1.0131 | 1.0154 |
| 15 | 14.993 | 1.0299 | 1.0352 |

The empirical ratio is consistently a little below the predicted one (e.g. $1.0299$ against $1.0352$ at $c \approx 15°$), which is the expected direction of discrepancy: the vertex-distance proxy measures a single discrete hexagon's actual finite boundary, not an infinitesimal circle, and a real aperture-7 hex cell also carries the fixed grid-alignment rotation discussed in Section 36.1 (not independently re-verified here at the level of individual cell orientation, only argued not to affect the ratio). The two columns track each other closely enough (within 2–5% relative, narrowing toward the center) to support the derivation without claiming an exact match at the single-cell, finite-resolution level.

**To Complete / Verify**

This chapter derives and verifies the *gnomonic component* of H3's projection in closed form, and argues – from the structure of the remaining steps, not from an independent re-derivation of them – that the fixed per-face azimuthal reference and the $19.10661°$ Class III rotation cannot affect $\omega$. It does not independently re-verify that argument against H3's actual digit-assignment code (`_hex2dToVec3` and neighbors), only against the closed-form mathematics of what a position-independent rotation and uniform scaling do to an ellipse. It also does not attempt DGGRID or OpenEAGGR: DGGRID's ISEA7H grid is a different, independently-implemented realization of the same conceptual (icosahedral, aperture-7) construction, and OpenEAGGR's ISEA3H uses a different aperture (3) and a different underlying equal-area (not gnomonic) polyhedral projection, reported in Snyder (1992)-family terms elsewhere in this volume; neither is re-derived here. Chapter 113 takes up a related but distinct question – not the local shape of each scheme's distortion field, but how their resolution indices relate to one another at a global, area-normalized scale.

Chapter 37

# Dutton's Quaternary Triangular Mesh: A DGGS Before the Acronym

## 37.1 Scope: an instrument used, not examined

Dutton's Quaternary Triangular Mesh (QTM) already has a presence in this volume, but only an instrumental one. Chapter 24 discussed Raposo et al. (2019)'s pedagogical virtual globe, built on QTM, as a tool for teaching the Modifiable Areal Unit Problem — but treated QTM there purely as substrate, the tessellation on which someone else's teaching device happened to be built, exactly the same move Chapter 21 made when it examined Fuller's Dymaxion map in isolation before Chapter 22 stepped back to place it in its own tradition. This chapter makes the same move for QTM: it goes back to Dutton's own founding paper and asks what QTM actually is, geometrically and historically, independent of the later uses this volume has already made of it.

**Methodological Note**

This chapter introduces two new sources to Section 114.6: Dutton (1989), Dutton's original AutoCarto 9 paper, whose full text was fetched and read directly in this pass; and Dutton (2016), a later retrospective by the same author, cited here only at the level of its published abstract, which was located via search but whose full text was not fetched in this pass — every claim attributed to it below is therefore restricted to what that abstract states explicitly. No numeric claim in this chapter rests on OCR-derived digits from scanned tables; where Dutton (1989)'s own statistics table was affected by scan-quality OCR noise, this chapter reports only the closed-form patterns the paper states explicitly in prose ($4^{\text{level}}$ facets and $2^{\text{level}}$ edge divisions per octant), not transcribed table entries.

## 37.2 Origins: a data-quality problem, not a tessellation problem

QTM's first public statement was a paper prepared for Specialist Meeting 1 of the newly formed National Center for Geographic Information and Analysis (NCGIA), held in Santa Barbara in December 1988, and published the following year in the proceedings of AutoCarto 9 under the title *Planetary Modelling via Hierarchical Tessellation* (Dutton, 1989). What is easy to miss, reading QTM only through its later citation history as "the octahedral DGGS that predates H3", is what problem Dutton actually opens the paper with: not tessellation efficiency, but locational data quality. His starting complaint is that GIS of the era recorded coordinates at uniform, high nominal precision regardless of how precisely those coordinates had actually been measured, and that no widely used system made a location's own precision an explicit, queryable property of the location itself (Dutton, 1989). QTM's hierarchical geocodes were proposed as a direct answer to that specific complaint: a facet address of length $L$ is, by construction, no more precise than the facet it identifies, so the number of digits in a QTM geocode is itself a state-

ment about how precisely that location was determined, independent of any separate metadata field (Dutton, 1989). The tessellation, in other words, was a means to an epistemic end — representing uncertainty as a first-class citizen of the coordinate system — rather than an end pursued for its own geometric interest. This motivation traces back further still: QTM is presented explicitly as the third stage of a lineage running from DEPTH (an elevation-encoding scheme using polynomial coefficients organised as quadtrees) through GEM, a global hierarchical triangular tessellation for terrain modelling, to QTM itself (Dutton, 1989), a progression corroborated independently, at the abstract level, by Dutton's own later retrospective account, which names the same three stages — DEPTH, GEM, QTM — in the same order (Dutton, 2016).

## 37.3 The geometry: octahedron, quaternary subdivision, ZOT

Geometrically, QTM starts from an octahedron rather than an icosahedron, the choice this volume has followed since Chapter 19 for the ISEA-family grids. Dutton's stated reason is orientation, not economy: an octahedron's vertices sit at right angles, so the solid aligns naturally with the cardinal directions of a geographic graticule, unlike the icosahedron, whose vertex placement (the subject of the entire disagreement examined in Chapter 21) has no comparably natural default (Dutton, 1989). Each of the octahedron's eight triangular faces subdivides recursively into four similar child triangles by connecting the midpoints of its edges, so that the number of facets per octant grows as $4^{\text{level}}$ and the number of edge subdivisions per side grows as $2^{\text{level}}$ (Dutton, 1989): each level of subdivision halves the linear resolution and quarters the facet area, the same aperture-4 growth law that Chapter 25 of this volume already treats in the context of H3 and rHEALPix, here appearing in the tessellation that first proposed it in a global DGGS context.

For computation, Dutton paired the spherical QTM tessellation with a planar projection of his own devising, the Zenithal OrthoTriangular (ZOT) projection, which maps the octahedron's eight faces onto eight right triangles packed into a square (Dutton, 1989). ZOT's purpose was purely computational economy: because its facets are right triangles in a regular grid, point-to-facet lookups (Dutton's own term for this is *trilocation*) can be resolved by comparing squared distances to candidate facet centroids, needing no square roots and, in the planar domain, no spherical trigonometry at all (Dutton, 1989).

## 37.4 Facet numbering, geocodes, and the attractor idea

QTM's numbering scheme is where its data-quality motivation becomes concrete. Each parent facet's four children are numbered 0 through 3: the central child, which points in the opposite orientation to its parent, is always 0; the three outer children, sharing the parent's orientation, are numbered 1, 2 and 3 by a fixed convention tied to the octahedron's poles and to the Greenwich meridian (Dutton, 1989). A QTM geocode is then simply the digit string recording which child was chosen at each level of recursive subdivision — Dutton's own worked example is the sixteen-level address `0311021223013032`, equivalent to the 32-bit hexadecimal number `3526B1CE`, which he notes carries the same order of positional resolution as a single Landsat pixel (Dutton, 1989). The property that makes this a genuine precision encoding rather than merely a compact address is that appending trailing zeros to a geocode never changes the location it denotes, only the precision claimed for it: `031`, `0311` and `03111` all denote the same underlying point at successively higher stated precision, whereas appending any other digit moves to a genuinely different, smaller facet (Dutton, 1989).

Dutton calls the shared vertex of a cluster of same-numbered facets an *attractor*: because facets numbered 1, 2 and 3 (but not 0) share a grid node with their same-numbered neighbours at the same level, the 1-, 2- and 3-facets across the whole tessellation organise themselves into hexagonal clusters of six triangles around each such node, while the 0-facets are left over to fill the gaps between clusters (Dutton, 1989). On his own accounting this hexagonal attractor pattern covers three-quarters of the tessellated surface, with the remaining quarter — the isolated 0-facets — filling the rest (Dutton, 1989). The resemblance to the hexagonal binning tradition examined at length in Chapter 73 of this volume is worth flagging explicitly: QTM is a triangular, not a hexagonal, tessellation, yet its own internal structure generates hexagonal groupings as an emergent by-product of a numbering scheme designed for an entirely different purpose (precision-bearing geocoding, not binning), rather than as a chosen cell shape.

## 37.5 QTM in the later DGGS landscape

QTM's own direct technical descendants — Dutton's later "hierarchical coordinate system for geoprocessing and cartography" papers, and applications such as Raposo et al. (2019)'s teaching tool already discussed in Chapter 24 — sit alongside, rather than inside, the lineage this volume has followed through Chapters 11, 12, 15 and 16: H3, S2, DGGRID's ISEA-family grids, and rHEALPix. QTM predates every one of them as a named, published, octahedral discrete global grid with an explicit hierarchical addressing scheme. It shares with the ISEA-family grids treated elsewhere in this volume the aperture-4 quadtree subdivision law of Section 37.3, and it anticipates, in its attractor construction, the same tension between triangular subdivision and hexagonal locality that later, independently motivated designs (H3 chief among them) resolve by simply choosing hexagonal cells from the outset.

**To Complete / Verify**

This chapter does not attempt an independent geometric reproduction of QTM comparable to the ones this volume has built for H3 (Chapter 12), DGGRID (Chapter 15) or EAGGR (Chapter 16): no IPQ, area-uniformity or resolution-extension statistics for QTM are computed or reported here. Doing so would require either an independent from-scratch implementation of the trilocation algorithm described in Section 37.3, or a verified existing package; Raposo (2019, GitHub) is a possible candidate implementation but was not evaluated in this pass, and no computational claim about it is made. This is left as an open item, in the spirit of the closing ledger of Chapter 114.

## 37.6 An independent geometric reproduction, and a numeric check of the attractor claim

The gap flagged in the todobox above is closed here, in the same spirit Chapter 12 adopted for H3 and Chapter 39 for HTM: not by evaluating Raposo (2019)'s existing package, but by implementing QTM's own aperture-4 octahedral subdivision from scratch (`code/qtm_independent_reproduction.py`, full output in `data/qtm_independent_reproduction_results.json`) and checking both its shape statistics and Dutton's own qualitative attractor claim (Section 37.4) against direct computation.

**Methodological Note**

The subdivision geometry itself — an octahedron recursively quartered by connecting spherical edge midpoints — is not re-derived independently in this pass: it reuses, unchanged, the same geodesic-midpoint construction already built from scratch for HTM in Chapter 39 (`code/htm_independent_ reproduction.py`), since Dutton (1989)'s own description of QTM's subdivision rule (Section 37.3) is the same "connect the edge midpoints" operation. What is new in this section is the QTM-specific content built on top of that shared geometry: facet-type (attractor) tagging and the numeric claims tied to it. Dutton's own numbering convention assigns digits 1, 2 and 3 to the three same-orientation ("corner") children by a rule tied to the octahedron's poles and the Greenwich meridian (Section 37.4), but Dutton (1989) as read for Section 37.4 does not give that rule in enough detail to reproduce the specific 1-versus-2-versus-3 assignment independently. This section therefore verifies only the structural CENTER/CORNER distinction the attractor claim actually depends on (opposite versus same orientation as the parent facet), without guessing which of the three corner children Dutton would call 1, 2 or 3.

**Shape statistics.** Because QTM's subdivision geometry is the one already built for HTM, the resulting shape statistics are identical, facet for facet, to Chapter 39's own figures: the maximum-to-minimum normalized-area ratio converges to approximately $2.106$ by depth 8, the normalized-area standard deviation stabilises at approximately $24.27\%$, and interior angles converge to the range $[45°, 90°]$ (Table 37.1) — the same asymptotic figures Chapter 39 reports for HTM, which is exactly what should be expected of two DGGS built on the same octahedral aperture-4 geometry, and is reported here as a positive cross-check rather than a coincidence requiring separate explanation.

Table 37.1: Independently computed QTM shape statistics by subdivision depth. Full table (depths 1–8) in `data/qtm_independent_reproduction_results.json`.

| Depth | $n$ facets | Area ratio max/min | Norm. area std | IPQ mean |
|---|---|---|---|---|
| 1 | 32 | 1.622 | 23.32% | 0.643 |
| 4 | 2,048 | 2.096 | 24.27% | 0.577 |
| 8 | 524,288 | 2.106 | 24.27% | 0.576 |

**The attractor claim, checked numerically.** At subdivision depth $L$, every parent facet contributes exactly one CENTER child (Dutton's digit 0) and three CORNER children (digits 1, 2, 3), so the count split is trivially exactly $3{:}1$ at any depth by construction — confirmed directly: $75.0\%$ of facets are CORNER type at every depth checked (Table 37.2). The area split is not trivial, since facets are not all equal in area, and converges toward the same $3{:}4$ figure from below as depth increases: $72.6\%$ of the surface by CORNER-facet area at depth 2, rising to $74.96\%$ by depth 5 — consistent with Dutton's stated "three-quarters of the tessellated surface" (Dutton, 1989) to within the precision his own prose states it.

The vertex-degree check is the more specific structural claim, and it holds exactly, with one clean exception. Grouping CORNER facets by the one parent vertex each retains, every shared vertex created by subdivision (i.e. every vertex other than the octahedron's own six original vertices) has *exactly six* CORNER facets around it at every depth checked (2 through 5) — the hexagonal cluster Dutton describes. The exception is at the octahedron's own six original vertices (the two poles and the four equatorial points): these have graph degree $4$, not $6$, at every

depth, because a vertex's degree in a triangulation built by recursive edge-bisection is fixed at construction and never changes under further subdivision, and the octahedron's own vertices have degree 4 to begin with. Dutton's own text, as cited in Section 37.4, does not flag this exception explicitly; it is a small but concrete instance of the same pattern already familiar from geodesic-dome and icosahedral DGGS constructions, where a handful of original-polyhedron vertices remain permanently singular against an otherwise uniform hexagonal (or, for QTM, attractor-triangle) tiling.

Table 37.2: Numeric check of Dutton's attractor claim: CORNER-facet fraction by count and by area, and vertex degree of CORNER-facet clusters. Full output in `data/qtm_independent_reproduction_results.json`.

| Depth | CORNER % (count) | CORNER % (area) | Degree, new vertices | Degree, original vertices |
|---|---|---|---|---|
| 2 | 75.0% | 72.55% | 6 (all) | 4 (all) |
| 3 | 75.0% | 74.40% | 6 (all) | 4 (all) |
| 4 | 75.0% | 74.85% | 6 (all) | 4 (all) |
| 5 | 75.0% | 74.96% | 6 (all) | 4 (all) |

**Resolution extension.** A separate check confirms that point-location (Dutton's own trilocation, reproduced here with the same squared-plane-membership test his paper describes as needing no square roots) is a genuine refinement across depths: for 15 arbitrary unit vectors, located independently at every depth from 0 to 8, the digit path found at depth $d + 1$ is, in all 15 cases and at every depth, the depth-$d$ path with exactly one digit appended — never a jump to a different branch of the tree. This is the property that makes Dutton's precision-bearing geocodes (Section 37.4) coherent across resolutions: refining a location's stated precision by adding digits never silently relocates it.

**To Complete / Verify**

This section verifies QTM's own aperture-4 shape statistics and its attractor structure, and reproduces trilocation-based point location and its resolution-extension property. It does not reproduce Dutton's specific 1-versus-2-versus-3 digit assignment among corner children (tied to a pole/meridian convention not given in sufficient detail in the source consulted for Section 37.4), nor his ZOT projection's planar packing of the eight octants into a unit square, which remains a separate, purely cartographic construction left unexamined here.

## 37.7 What Dutton's own framing adds to this volume's argument

The point worth carrying forward from this chapter is narrower than a full geometric characterisation of QTM, and is instead about framing. Chapter 1 opened this volume by asking what changes when a continuous distortion problem is replaced by a discrete tessellation one; the DGGS literature surveyed since then, this volume included, has mostly treated that replacement as a purely geometric substitution — swap a projection formula for a cell complex, and ask how uniform the cells are. Dutton's own 1988–1989 framing of QTM was different: for him, the discrete tessellation was valuable specifically because it let a location's precision travel with its coordinates, a property no continuous coordinate system offers as cleanly. That a data-quality argument, rather than a distortion-minimisation argument, motivated the first

octahedral DGGS in this volume's lineage is a detail the later, purely geometric citation history of QTM — including this volume's own use of it in Chapter 24 — has largely left behind.

Chapter 38

# The Hierarchical Triangular Mesh: The Same Octahedron, Independently, from the Sky

## 38.1 Scope: a second octahedron, from an unconnected literature

Chapter 37 examined Dutton's Quaternary Triangular Mesh as a named, published, octahedral discrete global grid whose earliest statement predates every ISEA-family or S2-family grid this volume has reproduced. This chapter does not revisit QTM's own geometry, which Chapter 37 already covered in full; instead it follows a second, independent line that arrives at the same root solid and the same subdivision rule from a completely different literature — astronomical database indexing rather than cartographic GIS — and asks what the two lineages, read side by side, say about the shape of the design space itself.

**Methodological Note**

This chapter introduces two new sources to Section 114.6: Kunszt et al. (2001), the earliest formal, named publication of the Hierarchical Triangular Mesh, cited here only at the level of its publisher-verified bibliographic record and abstract (full text is behind a Springer paywall and was not fetched in this pass); and Szalay et al. (2007), the fuller algorithmic treatment, whose abstract and submission metadata were fetched directly from arXiv in this pass, but whose specific numeric statistics reported in Section 38.4 below (the area-ratio and variance figures, and the angle bounds) were obtained via indexed excerpts of the paper's own body text returned by search rather than by a successful direct fetch of the full PDF, which returned only abstract-level content in every attempt made in this pass. This distinction is stated explicitly, following this volume's existing practice of separating material verified by full direct fetch from material available only as search-indexed excerpts (compare the treatment of Kmoch et al., 2022 in Chapter 1). No independent geometric reproduction of HTM is attempted; see the `To Complete / Verify` box in Section 38.5.

## 38.2 Origins: range queries on the sky, not precision on the ground

The Hierarchical Triangular Mesh first appears under that name in a short contribution by Kunszt et al. (2001) to the *Mining the Sky* symposium proceedings, motivated by a problem with no obvious connection to Dutton's 1988–1989 concern for locational data quality (Section 37.2): how to index hundreds of millions of objects in an astronomical catalogue — the Sloan Digital Sky Survey (SDSS) chief among them — so that a query for "every object within this region of sky" can be answered by an ordinary relational database's existing B-tree machinery, rather than by a bespoke spatial access method (Kunszt et al., 2001; Szalay et al., 2007). The fuller algorithmic treatment, Szalay et al. (2007), states this design goal explicitly as a com-

parative claim against continuous cartographic coordinate systems: an HTM-based index is presented as superior to coordinate-based approaches specifically because ordinary latitude/-longitude coordinates carry a singularity at the poles, which a hierarchical tessellation avoids by construction (Szalay et al., 2007). Where Dutton's motivating complaint was that no coordinate system made a location's own measurement precision explicit, Szalay, Gray, Fekete, Kunszt, Kukol and Thakar's motivating complaint was that no coordinate system let a relational database engine's own indexing machinery answer a spherical region query efficiently. Two unrelated problems, arriving twelve years apart, at the same architectural answer.

The astronomical side of this story has its own two-step precursor lineage, structurally parallel to the DEPTH-to-GEM-to-QTM lineage Chapter 37 traced for Dutton. Szalay et al. (2007) themselves credit an earlier application of a linear quadtree to astronomical databases, due to Barrett in 1994, as prior art within their own field, predating HTM's own naming by seven years (Szalay et al., 2007). As with Dutton's own DEPTH and GEM, this volume has not independently verified Barrett's 1994 paper (its full bibliographic record was not completed in this pass) and reports only that Szalay et al. (2007) themselves cite it as their immediate precedent.

## 38.3 The geometry: octahedron, aperture-4, trixels

HTM's construction is, at the level of the base solid and the subdivision rule, the same construction Section 37.3 described for QTM: start from an octahedron, treated as level 0, and recursively subdivide each triangular face into four children by bisecting its edges, generating what the HTM literature calls *trixels* (Szalay et al., 2007). Each trixel at a given level receives a hierarchical name built by appending a digit 0–3 to its parent's name, rooted in one of the eight level-0 octants (conventionally labelled by a hemisphere letter, N or S, and an octant digit); the deepest, most persistently central trixels at any give resolution carry names built from long runs of the same repeated digit (Szalay et al., 2007). This is the identical aperture-4 octahedral quadtree this volume has already met as QTM's own subdivision law in Section 37.3, arrived at without, on the evidence available in this pass, either lineage citing the other.

## 38.4 Shape statistics, as HTM's own literature reports them

Unlike QTM's founding paper, which stopped at the subdivision rule and the addressing scheme without reporting facet-shape uniformity statistics, the HTM literature reports its own quantitative shape diagnostics directly, in a form comparable to the IPQ and area-ratio statistics this volume has computed for H3, S2, DGGRID and rHEALPix in Chapters 11, 12, 15 and 16. Szalay et al. (2007) report that the ratio of the largest to the smallest trixel area at a given depth stabilises at approximately 2, that beyond depth 7 the distribution of relative trixel areas becomes self-similar with a scatter (variance) around the mean of approximately 24 percent, and that every trixel's interior angles stay bounded between $\pi/4$ and $\pi/2$ at every depth (Szalay et al., 2007). They further note that this shape irregularity originates entirely at the second level of subdivision, where the three corner trixels of each root octant are smaller than the central one, and that subsequent subdivisions neither eliminate nor amplify this initial asymmetry, only reproduce it self-similarly (Szalay et al., 2007).

**To Complete / Verify**

**Resolved in Chapter 39.** The figures above were transcribed, not recomputed, when this chapter was first written, subject to the sourcing caveat stated in this chapter's opening

disclaimer box. Chapter 39 closes that gap with a from-scratch implementation (no external HTM library), independently confirming and refining all three figures reported here (area ratio, area scatter, angle bounds). QTM's own shape statistics remain untested on QTM's specific address scheme, though Section 39.4 argues on structural grounds that they should coincide with HTM's, since both share the identical aperture-4 octahedral construction; this narrower point is still left open.

## 38.5 What the convergence, and not just the resemblance, suggests

The easy reading of Section 38.3 is that HTM and QTM merely resemble each other. The point worth making explicitly is stronger: two engineering communities with no evident contact between them — Dutton writing for the National Center for Geographic Information and Analysis in 1988, Szalay, Gray, Fekete, Kunszt, Kukol and Thakar writing for the Sloan Digital Sky Survey and Microsoft Research more than a decade later — independently selected the same base polyhedron and the same aperture-4 subdivision rule to solve two distinct problems (locational precision encoding versus relational range-query indexing) that share only a single structural requirement: a hierarchical, pole-free partition of the sphere addressable by short digit strings. This volume's own efficiency/distortion trade-off chapters (Part III) treat the proliferation of named DGGS schemes — H3, S2, DGGRID's ISEA family, rHEALPix, and now QTM and HTM — as points in a design space large enough to need the systematic comparison this volume has undertaken. The QTM/HTM convergence suggests that at least one region of that space, the octahedral aperture-4 quadtree, is not so much a design choice repeatedly made as a comparatively narrow attractor that two unrelated requirements happened to fall into independently. Whether the same can be said of the icosahedral aperture-4 family this volume has followed since Chapter 19 — whether, that is, an unrelated field converged on the icosahedron for reasons as independent as astronomy's route to the octahedron — is not examined here and is left as an open question for a future chapter.

Chapter 39

# An Independent Reproduction: HTM Geometry from Scratch

## 39.1 Scope: closing the item Chapter 38 left open

Chapter 38 transcribed three quantitative shape statistics from Szalay et al. (2007) — a max/min trixel-area ratio stabilising near 2, a relative-area scatter stabilising near 24 percent beyond depth 7, and interior angles bounded within $[\pi/4, \pi/2]$ — and left a `To Complete / Verify` box open, noting that no independent computation had been performed and that the natural extension would be to compute HTM's own normalised-area and ipq statistics directly, the way this volume already had for H3 (Chapter 12), DGGRID (Chapter 15), and OpenEAGGR (Chapter 16). This chapter does that, built entirely from scratch: no HTM library was used (none was identified or evaluated in the earlier pass), only the aperture-4 octahedral construction Section 38.3 already described in words.

**Methodological Note**

**Method.** An HTM builder was implemented from the description in Section 38.3: eight root trixels (four northern octants N0–N3, four southern S0–S3) on the octahedron with vertices at the poles and at longitudes $0°, 90°, 180°, 270°$ on the equator; each subsequent level bisects every edge at its great-circle midpoint (chord midpoint of the two endpoint unit vectors, renormalised to the unit sphere) and forms four children, the standard aperture-4 rule. Area and perimeter are computed *not* with the lon/lat-chart spherical-excess formula used for H3 in Chapter 12 (Chamberlain and Duquette 2007), but with the chart-free spherical-triangle-excess theorem, area $= (A + B + C - \pi)R^2$ from the three interior angles computed by dot products alone. This choice was forced by a bug caught in this pass, not a stylistic preference: every root trixel touches a pole, and half of them (this script's N2/S2 octants) have their opposite edge running exactly along the $\pm 180°$ meridian, a chart singularity the lon/lat formula does not survive (an outlier area roughly $20\times$ too large first flagged the problem; Section 39.2 below states this as a disclosed limitation of the earlier method, not a hidden correction). Full code: `code/htm_independent_reproduction.py`; full output: `data/htm_independent_reproduction_results.json`.

## 39.2 A caught error, stated rather than hidden

The first run of this reproduction, using the same polygon-area routine already validated for H3, reported a max/min trixel-area ratio of $49.6$ at depth $1$ and a ratio growing roughly fourfold per additional depth, reaching $3.4 \times 10^6$ by depth $9$ — nowhere near Szalay et al. (2007)'s reported figure of 2, and structurally implausible on its face, since an aperture-4 subdivision with a genuinely stabilising per-generation ratio cannot also have a globally compounding one

without contradiction. Tracing the outlier to a single root trixel (N2) confirmed the cause: two of its three vertices sit at longitude exactly $180°$ and $-180°$ (the same point, opposite chart branches), and the lon/lat spherical- excess formula, which sums a longitude-difference term edge by edge, picks up a spurious near-hemisphere contribution at that seam. This is disclosed here, rather than silently patched, in keeping with this volume's practice elsewhere (Section 38.5) of reporting what a first pass got wrong, not only what a later pass got right. Switching to the chart-free interior-angle formula (which never constructs a longitude at all) removed the discrepancy entirely; see Table 39.1 below.

## 39.3 Results: three published figures, checked independently

Table 39.1: HTM's octahedral aperture-4 construction, depths 1–9: max/min normalised-area ratio, relative-area scatter $\sigma$, mean ipq, and interior-angle range, all computed independently from a from-scratch implementation. Source: `code/htm_independent_reproduction.py`, `data/htm_independent_reproduction_results.json`.

| depth | #trixels | bits | ratio max/min | $\sigma$(norm. area)% | ipq mean | min angle | max angle |
|---|---|---|---|---|---|---|---|
| 1 | 32 | 5 | 1.6222 | 23.316 | 0.6428 | 54.736° | 90° |
| 2 | 128 | 7 | 1.9624 | 24.211 | 0.5922 | 47.266° | 90° |
| 3 | 512 | 9 | 2.0683 | 24.264 | 0.5800 | 45.556° | 90° |
| 4 | 2,048 | 11 | 2.0964 | 24.268 | 0.5770 | 45.138° | 90° |
| 5 | 8,192 | 13 | 2.1035 | 24.268 | 0.5763 | 45.035° | 90° |
| 6 | 32,768 | 15 | 2.1053 | 24.268 | 0.5761 | 45.009° | 90° |
| 7 | 131,072 | 17 | 2.1058 | 24.268 | 0.5761 | 45.002° | 90° |
| 8 | 524,288 | 19 | 2.1059 | 24.268 | 0.5760 | 45.001° | 90° |
| 9 | 2,097,152 | 21 | 2.1059 | 24.268 | 0.5760 | 45.000° | 90° |

**Key Result**

All three of Szalay et al. (2007)'s transcribed figures are independently confirmed, and refined. The max/min area ratio does not merely approach "approximately 2" but converges to $2.1059$ by depth 7 and does not move at all between depth $8$ and depth $9$ — genuine numerical stabilisation, not an artefact of stopping early. The relative-area scatter converges to $24.268$ percent, matching the reported "approximately $24$ percent" to within the precision that qualifier allows, and is already flat to three decimal places by depth $6$ – one depth earlier than Szalay et al.'s own stated threshold of depth 7, though this is a difference of degree, not of kind: both readings place stabilisation in the same narrow band of depths. The interior-angle bound is confirmed exactly, not merely approximately: the minimum angle converges to $45.000° = \pi/4$ and the maximum sits at exactly $90° = \pi/2$ at every depth tested, matching $[\pi/4, \pi/2]$ to the last digit reported.

## 39.4 Where this sits in the volume's wider comparison

Against the ipq figures this volume has already computed for the four polyhedral DGGS families it reproduces from scratch elsewhere — H3 and S2 (Chapter 11), DGGRID (Chapter 15), rHEALPix (Chapter 97) — HTM's converged mean ipq of $0.576$ is markedly lower than any hexagonal or square-cell scheme this volume has measured, all of which sit at $0.85$ or above once past their coarsest resolution. This is not a defect specific to HTM's implementation; it

is close to the theoretical ipq of an equilateral planar triangle ($\text{ipq} = \pi/(3\sqrt{3}) \approx 0.6046$), the shape every trixel approaches at fine resolution, and confirms from first principles what Section 38.5 already argued qualitatively: a triangular aperture-4 tessellation trades area uniformity (the ratio-to-2.1 result above is competitive) for a fixed, geometry-level compactness ceiling no refinement can raise, the same shape-versus- uniformity trade-off Chapter 92 frames for the whole volume, now with a fifth data point on the shape axis.

**To Complete / Verify**

This reproduction covers area, compactness and angle statistics only. It does not implement HTM's neighbour-finding algorithm (adjacency across trixels belonging to different root octants), which Chapters 106 and 107 disclose as an open item where it becomes relevant to timing. It also does not attempt QTM's own from-scratch reproduction on the identical octahedral rule (Chapter 37 transcribes QTM's founding paper without independent computation, exactly as Chapter 38 did for HTM before this chapter); since QTM and HTM share the same aperture-4 octahedral construction, the statistics in Table 39.1 above should apply to QTM unchanged, but this identity is asserted here on structural grounds, not independently re-verified for QTM's own address scheme, and is left for a future pass.

Chapter 40

# Closing the Gap, Differently: The Indicatrix Attempted, and Found Wanting, for HTM

## 40.1 Scope: why HTM cannot simply reuse Chapter 36's method

Chapter 36 closed the H3 gap by deriving, in closed form, the single azimuthal gnomonic tangent-plane projection H3's own source uses to build every one of its icosahedron faces. HTM has full computational-efficiency coverage in this volume (Chapters 106 and 107) but no indicatrix chapter at all – an asymmetry against every other grid this volume treats this way, and the gap this chapter closes. It cannot, however, be closed by the same method, for a structural reason disclosed here before any computation: Chapter 39's own from-scratch builder constructs HTM by repeated *chord-midpoint* bisection – each new vertex is the average of its two parent vertices, renormalized to the unit sphere – and every level re-anchors this operation to whichever three points are currently on the sphere, not to one fixed plane tangent at the root octant's centre. This is a genuinely different construction from projecting a flat, recursively bisected triangle through a single gnomonic chart, and whether the two nonetheless agree is an empirical question this chapter answers, not an identity assumed from the superficial resemblance between "eight triangular faces, recursively subdivided" and "twenty triangular faces, gnomonically projected."

**Methodological Note**

**What was done, precisely.** Two objects are computed and then compared, not conflated. First, the closed-form Tissot solution of a single azimuthal gnomonic projection tangent at root octant N0's own centroid – mathematically the identical object Section 36.2 already solved ($\omega = 2\arcsin\left(\frac{\sec^2 c - \sec c}{\sec^2 c + \sec c}\right)$, area scale $= \sec^3 c$), re-anchored here at a different point and valid out to a much larger circumradius ($54.736°$, computed directly from the same three real root vertices Chapter 39 uses – $(0,0,1)$, $(1,0,0)$, $(0,1,0)$ – not $37.377°$ as for an icosahedron face). Second, real HTM trixels, generated by *reusing, not reimplementing*, Chapter 39's own verified builder (`code/htm_independent_reproduction.py`: `NP`, `EQ`, `midpoint`, `point_in_trixel`, `spherical_polygon_area_perimeter`), located at controlled target directions by descending the real recursive tree to depth 11 (trixel edge $\approx 0.044°$ near the root, stable to at least four significant figures against depths 7 and 9; depth 13 begins to show floating-point drift from repeated normalize operations and is not used). Two real problems surfaced and were fixed rather than routed around: bearing $0°$ from the octant centroid runs exactly along the pole-angle bisector, an internal symmetry axis of the octant, and hit exact ties in the recursive point-in-triangle test at almost every level (multiple, or zero, of a level's four children claiming the same point); and a naive nearest-centroid fallback for points that had genuinely left the root octant silently returned a wrong trixel instead of stopping. Both are disclosed in Section 40.2 rather than patched

invisibly. Bearings $20°$ and $95°$, which avoid every symmetry axis of an equilateral spherical triangle, are used instead; each transect is truncated exactly where the target point is confirmed (`point_in_trixel` against the real root vertices) to have left octant N0, not carried past that point by any fallback. Full code: `code/tissot_htm_from_real_trixels.py`; full output: `data/tissot_htm_from_real_trixels_results.json`.

## 40.2 Two bugs caught in building the comparison

Both problems were caught by the same discipline Chapters 16 and 111 already applied elsewhere in this volume: run the real code on real positions and look at what it actually returns, rather than trust that a plausible-looking descent routine handles every case. The symmetry-axis tie is a direct consequence of octant N0 being an equilateral spherical triangle (all three sides and all three angles here are exactly $90°$, since a root octant is exactly one-eighth of the sphere): every recursive bisection splits it into four congruent children with a three-fold-symmetric arrangement, so a point sitting exactly on any of the octant's own three symmetry axes (through a vertex, or through an edge midpoint) lands exactly on a child-boundary edge at every subsequent level, not strictly inside any one child – a genuine structural feature of an equilateral triangle's recursive subdivision, not a coding slip, and the reason the two bearings used below were chosen specifically to avoid it. The second problem, silently mis-locating points that had already left the root octant, was a coding slip: the original descent routine picked whichever of a level's four children had the nearest centroid to the target point even when none of them, nor the root itself, actually contained it, producing a plausible-looking but meaningless trixel rather than an error. The fix was to check root membership before descending at all, and to stop each transect there rather than continue past it.

## 40.3 Near the centre: rough agreement

Table 40.1: HTM octant N0, real trixels (depth 11) versus the closed-form gnomonic comparison object, two bearings from the real centroid. Edge ratio is each real trixel's own longest/shortest great-circle side; $\sec c$ is the closed form's predicted value of the same ratio, evaluated at the real trixel's own actual angular distance from the centroid ($c_{\text{actual}}$, which tracks the target distance closely but not exactly, since the discrete tree does not offer a trixel centred at an arbitrary target). Source: `code/tissot_htm_from_real_trixels.py`, `data/tissot_htm_from_real_trixels_results.json`.

| $z$ target (°) | $c_{\text{actual}}$ (°) | edge ratio empirical | $\sec c$ predicted | bearing |
|---|---|---|---|---|
| 0 | 0.000 | 1.0000 | 1.0000 | both |
| 5 | 5.023 / 5.001 | 1.0088 / 1.0085 | 1.0039 / 1.0038 | 20° / 95° |
| 10 | 9.989 / 10.023 | 1.0350 / 1.0357 | 1.0154 / 1.0155 | 20° / 95° |
| 15 | 15.000 / 14.997 | 1.1163 / 1.1184 | 1.0353 / 1.0353 | 20° / 95° |

The two independent bearings agree closely with each other at every row (e.g. $1.1163$ vs. $1.1184$ at $15°$) – confirming, as directly as the earlier gnomonic chapter's own two-face check did for H3, that whatever is being measured here does not depend on which direction was walked, only on distance from the centre, at least this close in. Agreement with the gnomonic closed form itself is only rough and gets worse quickly: the empirical edge ratio already runs notice-

ably ahead of sec $c$ by 10–15° (1.116–1.118 measured against 1.035 predicted), roughly a 3–8% gap that is not shrinking as $c$ decreases toward 0 the way a genuine agreement in the limit would show – unlike Chapter 36's own empirical check, where real H3 cells tracked the closed form to within a few percent and the gap narrowed toward the centre.

**Key Result**

Real HTM trixels do *not* converge to the single-gnomonic-chart comparison object even close to the octant centre, and the gap widens with distance rather than the two curves meeting in the limit. This is the structural difference Section 40.1 predicted, now measured rather than assumed: repeated chord-midpoint bisection, re-anchored at every level to the current on-sphere vertices, is a genuinely different local operation from a single fixed tangent-plane projection, even infinitesimally close to the tangent point itself.

## 40.4 Farther out: area shrinks, not grows

Table 40.2: Real trixel area, normalized to the centroid trixel's own area, versus the closed form's predicted area scale ($\sec^3 c$). Source: as Table 40.1.

| $z$ target (°) | area / centre-area empirical (bearing 20°) | area scale predicted ($\sec^3 c$) |
|---|---|---|
| 0 | 1.0000 | 1.0000 |
| 15 | 0.8059 | 1.1096 |
| 25 | 0.5218 | 1.3431 |
| 35 | 0.5628 | 1.8195 |
| 40 | 0.5646 | 2.2223 |

The predicted gnomonic area scale grows monotonically away from the tangent point, as Chapter 36 found for H3 and as any azimuthal gnomonic projection must. Real HTM trixels do the opposite over this same range: normalized area *falls* from 1.0 at the centre to about 0.52–0.56 by 25–40° out, with a real, disclosed non-smooth jump between 20° and 25° (confirmed independently at both bearings, so not a single stray measurement) most plausibly attributable to the discrete tree's own alternating "upward" and "downward" child orientation at that depth rather than a continuously varying field – a genuine feature of a discrete recursive triangular tessellation this chapter does not further decompose.

**Key Result**

HTM's recursive octahedral construction distorts area in the opposite direction from the natural single-gnomonic-chart comparison object, not merely by a different amount. Cells farther from a root octant's centre are measured here as smaller, not larger, than cells near it – consistent with, and now a candidate mechanism for, the max/min area ratio of 2.1059 and the 24.268% relative-area scatter Chapter 39 already reported at depth $\geq 7$, neither of which that chapter traced to a specific spatial pattern. This chapter supplies that missing spatial reading: those aggregate statistics are consistent with, at least in part, a real centre-to-edge area gradient of the opposite sign from every other polyhedral DGGS this volume has measured a gnomonic or ISEA-family construction for (Chapters 36 and 89), not independently re-derived as a closed-form law here, only measured and disclosed.

**To Complete / Verify**

Three things this chapter does not do. First, it does not derive a closed-form replacement for the naive gnomonic comparison object; having shown that comparison fails, it stops at the measurement rather than fitting a new curve to it, since curve-fitting eleven points to a several-parameter form this chapter has no independent reason to prefer would manufacture a false sense of having explained the mechanism. Second, the centre-to-edge area-shrinkage finding of Section 40.4 is reported only for octant N0 and only along two bearings; whether it holds for all eight root octants and all azimuths, or is itself a symmetry-axis-adjacent artefact of the two bearings chosen, is not checked. Third, this chapter does not attempt QTM, which Chapter 39's own todobox already flagged as sharing HTM's aperture-4 octahedral construction but not independently re-verified for its own address scheme; the same caution applies here with even more force, since this chapter's central finding is exactly that surface-level construction similarity ("recursively subdivided triangular faces") does not guarantee that two schemes' local distortion behaves the same way.

Chapter 41

# The Indicatrix, Side by Side: A Cross-Grid Synthesis

## 41.1 Scope: six chapters, one comparison never assembled

By this point in the volume, six different grids have each had a real Tissot indicatrix – or, for one of them, a real and disclosed failure to obtain one by the standard method – computed from a real source: S2 and rHEALPix from their own continuous forward transforms (Chapter 31), H3 from a closed-form derivation of its gnomonic construction (Chapter 36), DGGRID and OpenEAGGR from the actual compiled DGGRID binary plus a term-by-term source comparison (Chapter 89), HTM from real trixels generated by its own verified builder (Chapter 40), and GBT from the exact planar construction already established elsewhere in this volume (Chapter 110). No chapter has yet placed these six results side by side. This one does – and only that: no new library call, no new script, and no new numerical claim is made here that is not already a table entry or a keyresult in one of the six source chapters just listed.

**Methodological Note**

**What this chapter is, and is not.** This is a synthesis of already-published numbers, not a new measurement. Every figure below is copied, not recomputed, from the table or keyresult cited beside it. The six source chapters used six different methods – a closed-form analytic derivation (H3), a finite-difference Jacobian on a library's own continuous transform (S2, rHEALPix), finite-differencing a real compiled binary's text output (DGGRID/OpenEAGGR), an exact planar construction with no sphere involved at all (GBT), and an empirical edge-ratio and area-ratio comparison against real recursively-built cells after the standard indicatrix comparison was shown not to apply (HTM) – at different, not-directly-matched angular distances from each grid's own best-behaved point, using different definitions of that distance (great-circle angle from a face or octant centroid; degrees of latitude; a diagonal transect's own step parameter). The comparison that follows is therefore qualitative – which grids grow worse quickly versus slowly, which are monotonic versus not, which trade angular fidelity for exact area preservation and which do not – not a claim that any two entries at "the same" nominal distance are measuring identical geometric configurations.

## 41.2 The six figures, collected

## 41.3 What the six have in common, and where they genuinely differ

Three constructions here are exactly equal-area by design – rHEALPix, the shared DGGRID/OpenEAGGR ISEA map, and GBT – and all three leave angular deformation free to vary however the area constraint forces it to. Among them, GBT is the only one that reaches

an angular deformation of exactly zero, and it does so for a structural reason none of the other two can share: it is never projected onto a curved surface at all, so the usual trade-off this volume's cost-of-distortion chapters trace as a curve (Chapter 110's own framing) does not apply to it in the first place. rHEALPix and DGGRID/OpenEAGGR both pay for their exact area preservation with angular deformation in the same rough range at their respective worst points measured so far ($22.8°$ and $15.5°$), though neither chapter searched exhaustively for a true global maximum, so this is a comparison of the largest values each chapter happened to measure, not of two verified maxima.

H3 and S2 are the two constructions built around a single distinguished point of exact isotropy ($\omega = 0$) – a face centroid in both cases – with deformation that grows away from it. Both are consistent with an ordinary gnomonic-type projection's expected behavior once Chapter 32 corrected the one apparent counter-example (S2's reported fall near its own face's nominal corner, shown there to be a different face's Jacobian sampled by a transect that had already crossed a real face boundary, not a true reversal of face 0's own map). H3 was measured only out to $20.9°$ of its $37.377°$ face circumradius, so its own worst-point figure in Table 41.1 is a lower bound on the true face-corner value, not the value itself – a genuine gap this synthesis does not close, since doing so would mean extending Chapter 36's own transect, a new computation outside this chapter's remit.

HTM is the one construction in this table that resists being placed on the same axis as the other five at all. Chapter 40 did not merely find a larger or smaller $\omega$ for HTM; it found that the entire comparison object – a single tangent-plane gnomonic projection, the same kind of object that described H3, S2's own face map, and the shared ISEA construction adequately, if to different degrees – is the wrong kind of object for HTM's actual construction (repeated chord-midpoint bisection, re-anchored to the current sphere at every level) to converge to, even near the centre. Its area behavior is the more striking finding precisely because it is measurable without that contested comparison: real trixel area *falls* away from the octant centroid, where every other construction in Table 41.1 grows monotonically or, for the three equal-area schemes, holds exactly constant. No other grid measured in this volume behaves this way.

**Key Result**

Of six grid constructions with a real, source-derived distortion measurement in this volume, three are exactly equal-area by construction (rHEALPix, the shared DGGRID/OpenEAGGR ISEA map, GBT) and three are not (H3, S2, and – on the area axis specifically, in the opposite direction from the other five – HTM). Among the two non-equal-area schemes built around a single isotropic reference point (H3, S2), angular deformation grows monotonically away from that point once a genuine sampling artifact is corrected for (Chapter 32). HTM is the sole exception on both counts: no $\omega$ figure was obtainable by the method that worked for the other five, and its area scale moves in the opposite direction from every other grid measured, a finding Chapter 40 attributes to HTM's fundamentally different (re-anchored, chord-based rather than tangent-plane) recursive construction rather than to any failure of measurement.

**To Complete / Verify**

Three limits, disclosed rather than papered over. First, this chapter performs no controlled experiment: the six "worst point" figures in Table 41.1 were not measured at matched angular distances, matched sampling density, or even a common definition of distance from centre, so the table supports qualitative ordering (which grids are monotonic, which are exactly equal-area, which trade what for what) and not a ranked league table of single numbers. Second, several of the individual worst-point figures are themselves dis-

closed lower bounds on an unmeasured true extremum – H3's transect stopped at $20.9°$ of a $37.377°$ face, DGGRID's at $35°$ of the same $37.377°$ circumradius, and rHEALPix's own todobox already flagged its interior minimum near $\phi \approx 25°$ as unconfirmed against the full two-dimensional field – closing any of these would require extending the corresponding source chapter's own transect, not a new chapter of this kind. Third, QTM remains entirely outside every chapter cited here, exactly as Chapter 39's own todobox and Chapter 40's both already flagged: it shares HTM's aperture-4 octahedral construction closely enough that this chapter's finding – that surface-level construction similarity does not guarantee matching local distortion behavior – is itself a reason not to assume QTM inherits either H3-family or HTM-family behavior without measuring it directly.

Table 41.1: Angular deformation $\omega$ and area behavior, six grids, as already measured elsewhere in this volume. "Best point" is each construction's own most favorable location (a face/octant centroid or a design-nominated reference point), not a common global reference. Sources as cited in the rightmost column; no value here is recomputed.

| Grid | Construction | $\omega$ at best point / worst point measured | Area behavior | Source |
|---|---|---|---|---|
| H3 | gnomonic, per icosahedron face | $0°$ at face center / $3.899°$ at $c$=$20.9°$ (face circumradius $37.377°$, not reached) | grows monotonically, area scale $\sec^3 c$ | Table 36.1 |
| S2 | quadratic-corrected gnomonic cube | $0°$ at face center / $35.604°$ at step $44°$ (own-face closed form; auto-selected transect wrongly showed a fall to $20.748°$, resolved as a face-boundary sampling artifact) | rises with occasional local minima; own-face law is monotonic | Table 31.2, Table 32.1 |
| rHEALPix | equal-area cylindrical, quad + polar cap | $1.882°$ near $\phi$=$25°$ / $22.762°$ at $\phi$=$45°$ (cap boundary) | exactly equal-area everywhere (area scale constant to 7 s.f.) | Table 31.1 |
| DGGRID / OpenEAGGR | ISEA, per icosahedron face | $\approx$ $1.7°$ at Newton-located face center / $15.541°$ at $z$=$35°$, bearing $90°$ | equal-area, constant to $\pm 0.3\%$ across 16 points measured | Table 89.1 |
| HTM | chord-midpoint octahedral bisection | no $\omega$ obtained: standard gnomonic comparison shown not to apply (edge ratio $1.12$–$1.12$ vs. predicted $1.035$ at $c$$\approx$$15°$, gap widening not narrowing) | *shrinks* with distance from centre ($1.00 \rightarrow 0.52$–$0.56$ by $25$–$40°$), opposite every other grid here | Table 40.1, Table 40.2 |
| GBT | unbounded planar hexagonal tiling | not a Tissot indicatrix quantity: $\sigma$(norm. area) $= 0$ and ipq $= 0.90690$ exactly, at every level, by construction | never projected onto a sphere; distortion floor is exact, not approached | Table 110.1 |

Chapter 42

# Not an Artefact of One Octant: The HTM Transect, Extended

## 42.1 Scope: the second item of Chapter 40's todobox

Chapter 40 measured, along two bearings from root octant N0's centroid, that real HTM trixels neither converge to the single-gnomonic-chart comparison object near the centre nor grow in area away from it the way that comparison object does – they shrink. Its own todobox left open whether this holds for all eight root octants and a wider spread of azimuths, or is instead an artefact of octant N0 and the two bearings $(20°, 95°)$ originally chosen to avoid that octant's symmetry axes. This chapter closes that item.

**Methodological Note**

**What was done, precisely.** Chapter 39's own from-scratch builder (`code/htm_independent_reproduction.py`: `NP`, `SP`, `EQ`, `ROOT_TRIXELS` – all eight root octants, not only N0 – `midpoint()`, `point_in_trixel()`, `spherical_polygon_area_perimeter()`) is reused unchanged. A new transect/-comparison harness generalizes Chapter 40's single-octant script to all eight octants and six candidate bearings per octant $(20°, 80°, 140°, 200°, 260°, 320°$, measured from a local tangent-plane frame whose $0°$ direction is the projection of the octant's own pole vertex onto the tangent plane at the octant's centroid), at nine target angular distances ($0°$ to $40°$ in $5°$ steps) and real recursive descent to depth 11, exactly as before. Symmetry-axis ties are not assumed away: every single descent, at every level from 1 to 11, is checked for an exact tie (zero or more than one matching child), and any occurrence is logged and excluded rather than silently resolved, the same discipline Section 40.2 applied to bearing $0°$. Full code: `code/htm_octant_bearing_sweep.py`; full output: `data/htm_octant_bearing_sweep_results.json`.

## 42.2 No ties, and an unplanned symmetry finding

Across all eight octants, six bearings, and nine target distances (432 transect points), zero exact ties occurred at any of the 11 descent levels – the six bearings chosen, all $20°$ off the $60°$-spaced symmetry axes every root octant has by construction (three $90°$-$90°$-$90°$ equilateral spherical triangles' own reflective symmetry axes, through each vertex and its opposite edge's midpoint), avoid the exact-tie case entirely at this depth. A second, unplanned finding fell directly out of the six-bearing design: because each root octant has exact three-fold rotational symmetry, bearings $20°/140°/260°$ are not three independent samples but three exact copies of the same underlying geometry, and likewise for $80°/200°/320°$ – confirmed numerically, not assumed, since every quantity computed below agrees between the three bearings of a class to at least six significant figures, at every octant. The six nominal bearings therefore supply exactly

two azimuthally distinct test directions per octant, not six – still a genuine widening of Chapter 40's original two bearings ($20°$, $95°$), since neither of this chapter's two classes coincides with the original $95°$ choice, and now replicated across all eight octants rather than reported for N0 alone.

## 42.3 Octant invariance

Every quantity in Tables 42.1 and 42.2 below is identical, to six significant figures, across all eight root octants (N0–N3, S0–S3) at matching bearing-class and target distance – confirmed directly, not assumed from the octahedron's textbook symmetry, by computing all eight independently and comparing. Root octant N0's own centre-trixel area ($25{,}142{,}470.12\,\mathrm{m}^2$, $R = 6{,}371{,}000\,\mathrm{m}$, cancels in every ratio used) matches N1, N2, N3, S0, S1, S2, S3 to within $2 \times 10^{-8}$ relative difference, consistent with floating-point noise rather than any real octant-to-octant asymmetry.

**Key Result**

Chapter 40's two core findings are not an artefact of octant N0 or of the specific bearings $20°/95°$: both the near-centre edge-ratio non-convergence and the farther-out area shrinkage reproduce identically across all eight root octants, at both azimuthally distinct bearing classes available within an octant's three-fold symmetry. The octahedral root tessellation is, as its construction guarantees, perfectly isotropic across octants; it is *not* perfectly isotropic across azimuth within a single octant.

## 42.4 Near the centre: the non-convergence, at two independent azimuths

Table 42.1: Empirical edge ratio versus the closed-form gnomonic prediction $\sec c$, at the two azimuthally distinct bearing classes (representative values; identical across all eight octants to six significant figures). Source: `code/htm_octant_bearing_sweep.py`.

| $z$ target (°) | edge ratio class A (20°) | edge ratio class B (80°) | $\sec c$ predicted | $c_{\text{actual}}$ (°) |
|---:|---:|---:|---:|---:|
| 5 | 1.0088 | 1.0091 | 1.0039 | 5.02 |
| 10 | 1.0350 | 1.0340 | 1.0154 | 9.99 |
| 15 | 1.1163 | 1.1118 | 1.0353 | 15.00 |
| 20 | 1.1207 | 1.1239 | 1.0642 | 20.00 |
| 25 | 1.2991 | 1.3233 | 1.1033 | 24.99 |
| 30 | 1.3202 | 1.3271 | 1.1548 | 30.01 |
| 35 | 1.2905 | 1.3108 | 1.2208 | 35.00 |

Both azimuth classes reproduce Chapter 40's finding: the empirical edge ratio runs ahead of $\sec c$ from as early as $10°$–$15°$ out, and the gap widens rather than narrows approaching the centre. The two classes agree with each other closely at every row (within a few parts per thousand through $20°$, diverging a little more from $25°$ on), extending Chapter 40's own bearing-independence check to a second, genuinely different azimuth relative to the octant's own vertex/edge-midpoint geometry, not merely a second arbitrary compass direction.

## 42.5 Farther out: area shrinkage, confirmed at both azimuths and every octant

Table 42.2: Real trixel area normalized to the centroid trixel's own area, against the closed form's predicted area scale ($\sec^3 c$), at both bearing classes (identical across all eight octants to six significant figures). Source: as Table 42.1.

| $z$ target (°) | area/centre-area class A | area/centre-area class B | area scale predicted ($\sec^3 c$) |
|---|---|---|---|
| 15 | 0.8059 | 0.8053 | 1.1096 |
| 20 | 0.8051 | 0.8039 | 1.2052 |
| 25 | 0.5218 | 0.5176 | 1.3431 |
| 30 | 0.5294 | 0.5171 | 1.5401 |
| 35 | 0.5628 | 0.5297 | 1.8195 |

An azimuth-dependent structural finding, not present in Chapter 40's two-bearing report, also fell out of this sweep: at target distance $40°$, every class-B transect (bearing class near an edge midpoint) had already left root octant N0 by the check against `point_in_trixel`, while every class-A transect (bearing class near a vertex) had not. This is a genuine, disclosed feature of an equilateral spherical triangle's own shape, not a script error: the octant's angular extent from its centroid is azimuth- dependent, larger toward a vertex ($54.736°$, per Chapter 40's own closed-form circumradius) than toward an opposite edge's midpoint, and this sweep is the first place in this volume that distance is measured, rather than only invoked as the single circumradius figure.

**Key Result**

The area-shrinkage finding of Chapter 40 reproduces identically across all eight root octants and both azimuthally distinct bearing classes available within an octant's symmetry: real HTM trixel area falls to roughly half the centre-trixel area by $25°$–$35°$ out, versus the naive gnomonic comparison object's predicted growth to $1.34$–$1.82\times$, confirming this is a structural property of HTM's recursive chord-midpoint construction, not an artefact of octant choice or of the original chapter's particular two bearings. The octant's own angular extent is itself azimuth-dependent (shorter toward an edge midpoint than toward a vertex), a genuine feature this sweep newly measures rather than one this volume previously computed.

**To Complete / Verify**

This chapter tested two azimuthally distinct directions per octant, set by the three-fold symmetry every root octant shares; it did not sweep a continuum of azimuths within the $60°$ gap between symmetry axes, so whether the edge-ratio and area figures vary smoothly between the two classes reported here, or do something non-monotonic in between, is not measured. Nor does this chapter revisit Chapter 40's own first todobox item – deriving a closed-form replacement for the naive gnomonic comparison object – for the same reason given there: eleven or eighteen points are not enough to prefer one several-parameter curve over another without an independent reason to do so, and this chapter's octant/azimuth sweep, however wider, does not supply that reason either.

Chapter 43

# Between the Two Points: Sweeping the Fundamental Domain

## 43.1 Scope: the first item of Chapter 42's todobox

Chapter 42 tested six discrete bearings per root octant ($20°$, $80°$, $140°$, $200°$, $260°$, $320°$), which its own three-fold rotational and reflective symmetry reduce to two azimuthally distinct representative angles within a $60°$ fundamental domain. Its own todobox left open whether the edge-ratio and area-shrinkage figures vary smoothly between those two representative points, or do something non-monotonic in between – a question six discrete samples cannot answer either way. This chapter sweeps the fundamental domain directly.

**Methodological Note**

**What was done, precisely.** Bearings $3°, 6°, \ldots, 57°$ (19 values, spaced $3°$ apart) are swept for every one of the eight root octants, at three polar-distance offsets $z \in \{10°, 20°, 30°\}$ – a reduced subset of Chapter 42's own nine $z$ values, chosen as the range where that chapter's discrete-bearing table showed the clearest separation from the naive gnomonic prediction, to keep this chapter's $8 \times 19 \times 3 = 456$-point sweep computationally proportionate to what a smoothness check requires. Bearings $0°$ and $60°$ themselves, the exact symmetry axes already disclosed as tie cases in Chapter 40, are excluded from the swept range by construction rather than caught after the fact. All geometry – `ROOT_TRIXELS`, `descend()`, recursive chord-midpoint bisection to depth 11, `spherical_polygon_area_perimeter()` – is imported unmodified from `code/htm_octant_bearing_sweep.py`; only the bearing list is new. Source: `code/htm_octant_continuous_azimuth_sweep.py`, `data/htm_octant_continuous_azimuth_sweep_results.json`. No ties were excluded at any of the 456 points swept.

## 43.2 Results

The answer is unambiguous and the same across all eight octants: the curve between the two previously-tested representative angles is *not* smooth. At $z = 10°$ (near the octant centroid) the series is close to smooth, with a maximum bearing-to-bearing relative change in the area ratio of only $1.1\%$ and a mean step of $0.13\%$. But at $z = 20°$ and $z = 30°$ (the range where Chapters 40 and 42's own findings are strongest), the area ratio jumps by as much as $55.0\%$ and $53.4\%$ respectively between adjacent $3°$ steps, against a mean step size around $3.5$–$3.7\%$ – an order of magnitude larger than the near-centre case.

Table 43.1: Sign changes in the first difference of the area ratio and edge ratio across the 19-bearing sweep (maximum possible: 17, for 18 consecutive differences), by polar offset $z$. Every one of the eight root octants gives the same count at each $z$ (shown once).

| $z$ | area-ratio sign changes | edge-ratio sign changes | max relative jump (area) |
|---|---|---|---|
| $10°$ | 10 (N0–N3, S0, S1) / 12 (S2, S3) | 14 | 1.1% |
| $20°$ | 13 | 8 | 55.0% |
| $30°$ | 10 | 3 | 53.4% |

**Key Result**

Real HTM trixel geometry does *not* vary smoothly with azimuth between the two representative bearings Chapter 42 tested: 270 sign changes in the area-ratio's first difference and 200 in the edge-ratio's, summed across the 24 (octant, $z$) series swept, most of them concentrated at $z \geq 20°$ where relative jumps of order 50% occur between adjacent $3°$ bearing steps. This is not measurement noise (the near-centre case at $z = 10°$ is smooth to within 1.1%, showing the method itself is not the source); it is real, physical consequence of the recursive chord-midpoint subdivision: as azimuth sweeps at fixed angular distance from the centroid, the target point crosses trixel boundaries within the depth-11 recursive tiling at different levels for different bearings, so a small change in azimuth can occasionally select a structurally different leaf trixel rather than a continuously deformed one. The finding is consistent across all eight root octants without exception, ruling out octant N0 (or any single octant) as a special case – the jaggedness is a generic property of HTM's construction, not an artefact of Chapter 40's original choice of octant.

## 43.3 What this rules in, and what remains open

This closes Chapter 42's first todobox item with a clear answer – non-monotonic, not smooth – while sharpening what that non-monotonicity means. It is not evidence against Chapter 40's central finding (real trixel area shrinks away from centre, contrary to the naive gnomonic prediction of growth); the shrinkage trend is visible in the coarse six-bearing sample specifically because six points, spaced widely, happen to average over the jaggedness this chapter now reveals at fine resolution. What the two-bearing (or six-bearing) sampling in the earlier chapters could not show, and this chapter's finer sweep does, is that any single-bearing measurement at $z \gtrsim 20°$ carries real, non-negligible azimuthal variance around that trend, up to several tens of percent from one $3°$ step to the next.

**To Complete / Verify**

This chapter used $3°$ steps, coarse enough that individual trixel-boundary-crossing events are inferred from sign changes in the first difference rather than located exactly. A finer sweep (sub-degree steps) restricted to a single $3°$ interval already shown to contain a large jump would locate the exact azimuth of a boundary crossing and confirm it coincides with a specific recursive-descent tie-breaking level, rather than assuming this from the aggregate pattern alone. Second, this chapter's reduced $z$ subset $(10°, 20°, 30°)$ does not cover Chapter 42's full nine-point range; whether the jaggedness continues to grow, plateaus, or attenuates beyond $z = 30°$ or below $z = 10°$ is not measured here.

Chapter 44

# One Hundredth of a Degree: Locating the Crossing

## 44.1 Scope: the first item of Chapter 43's todobox

Chapter 43 swept the fundamental domain in $3°$ steps and found the single largest jump anywhere in its $8 \times 19 \times 3$-point sweep at octant N1, $z = 20°$, between bearings $12°$ and $15°$: the area ratio moved from $0.5192$ to $0.8050$, a $55.0\%$ relative jump between adjacent samples. That chapter's own todobox left open whether a finer, sub-degree sweep restricted to this one interval would locate the exact crossing bearing and confirm it coincides with a specific recursive-descent level's tie-breaking decision, rather than assuming this from the aggregate $3°$-resolution pattern alone. This chapter runs that finer sweep directly.

**Methodological Note**

**What was done, precisely.** `ROOT_TRIXELS`, `NP`, `SP`, `build_local_frame()`, `target_at()`, `point_in_trixel()`, `spherical_polygon_area_perimeter()`, and `DEPTH=11` are imported unchanged from `code/htm_octant_bearing_sweep.py`. The one addition is `descend_with_path()`, a direct generalization of that module's own `descend()` that returns, in addition to the same tie/no-tie result, the full sequence of matched child indices (0–3; indices 0–2 are the three corner sub-trixels, 3 is the central sub-trixel) at each of the 11 recursive levels – the underlying point-in-trixel test and bisection are otherwise byte-for-byte identical. Octant N1 at $z = 20°$ is swept at 301 points across exactly the $[12°, 15°]$ interval Chapter 43 identified, a step of $0.01°$ – three hundred times finer than that chapter's $3°$ steps. Source: `code/htm_octant_subdegree_crossing_check.py`; full output: `data/htm_octant_subdegree_crossing_results.json`. Zero ties were excluded across all 301 points.

## 44.2 Results

The crossing is located precisely: the largest single-step jump in this 301-point fine sweep falls between bearing $13.74°$ and $13.75°$ – a $0.01°$ step, the finest resolution tested – where the area ratio jumps from $0.519203$ to $0.804798$, a $55.01\%$ relative change, matching Chapter 43's $3°$-resolution figure ($55.0\%$) to within rounding. The crossing bearing therefore lies inside the original $3°$ bracket almost exactly where a naive linear interpolation across that bracket would place it ($12° + (15° - 12°) \times \frac{13.74-12}{15-12} \approx 13.74°$), confirming the coarse sweep's jump was a single sharp event rather than several smaller compensating jumps that happened to sum to a similar total.

**Key Result**

Comparing the two descent paths straddling the crossing bearing ($13.74°$ and $13.75°$) level by level shows they first diverge at recursive level $0$ – the very first subdivision of the root octant itself, not a deep, fine-grained level as the smallness of the triggering bearing change ($0.01°$) might suggest. A target point $0.01°$ of arc away from another can therefore land in a structurally different one of the four level-$0$ children of the root trixel, because the point in question sits extremely close to that level's own bisecting geodesic; once the level-$0$ child differs, every subsequent level's geometry is computed relative to a different parent trixel, so the two paths are not expected to reconverge, and the resulting leaf trixels (and their areas) can differ substantially even though the two target points themselves are separated by only a hundredth of a degree.

Across the full $301$-point sweep, $36$ of the $300$ adjacent sub-degree pairs show a descent-path divergence at some level (the remaining $264$ pairs stay within the same leaf trixel throughout all $11$ levels, differing only by the smooth, continuous deformation of a single trixel's own shape). Of those $36$ divergent pairs, only the one identified above diverges at level $0$; the rest cluster at the deepest levels tested (level $9$: $9$ pairs; level $10$: $17$ pairs; levels $6$–$8$: $9$ pairs combined), consistent with most nearby bearings crossing only fine, late-stage trixel boundaries whose effect on total area is comparatively small, while the single largest jump in the whole coarse sweep happens to be the rare case of a level-$0$ crossing.

**Key Result**

This confirms, precisely as the todobox asked, that the $55\%$ jump Chapter 43 found at $3°$ resolution is not an artefact of that resolution or of measurement noise: it is a single, real, exactly locatable crossing of a specific recursive-descent decision boundary (level $0$ of $11$), located here to within $0.01°$ of arc. The other $35$ smaller divergences found in the same fine sweep occur overwhelmingly at deep levels, explaining why they do not individually produce jumps of comparable size to the one anomaly this chapter locates.

## 44.3 What this closes

This chapter closes the first item of Chapter 43's todobox: the exact crossing bearing is located to $0.01°$, and the level at which the descent path diverges is identified directly (level $0$) rather than inferred. That chapter's second item – extending the sweep beyond the reduced $\{10°, 20°, 30°\}$ subset to the full nine-point range Chapter 42 originally used – remains untested and is not attempted here, since it is a distinct question (coverage across $z$) from the one this chapter was scoped to answer (resolution at a single, already-identified $z$).

Chapter 45

# Generalized Balanced Ternary: An Addressing Algebra Seven Years Before QTM

## 45.1 Scope: a digit assignment cited, never examined

Generalized Balanced Ternary (GBT) has already left a mark on this volume without ever appearing under its own name in a chapter title. Chapter 25 took directly from Sahr et al. (2003) and Sahr (2011) the classification of 3, 4, and 7 as the three central-place apertures, and Chapter 12 built an independent reproduction of H3's aperture-7 hexagonal hierarchy; but the specific rule by which digits are assigned to the seven cells of an aperture-7 unit — which cell is called 0, which cells are called 1 through 6, and why that particular arrangement makes negation and vector addition cheap — was left as an unexamined input to those chapters, exactly the way Chapter 24 left QTM as an unexamined substrate before Chapter 37 went back to Dutton's own paper. This chapter makes the same move for GBT: it goes back to the system's own 1982 origin and its first detailed technical exposition, and asks what GBT actually is, independently of the later DGGS uses this volume has already made of it.

The historical point is worth stating plainly before the geometry: Generalized Balanced Ternary predates Dutton's QTM (1989, discussed in Chapter 37) by seven years, having been first described by Laurie Gibson and Dean Lucas at the 1982 IEEE Computer Society Conference on Pattern Recognition and Image Processing in Las Vegas (Gibson & Lucas, 1982). Unlike QTM, however, GBT was not proposed as a global terrestrial tessellation at all — a distinction this chapter's closing section returns to.

**Methodological Note**

This chapter introduces three new sources to Section 114.6. Gibson & Lucas (1982), Gibson and Lucas's original 1982 conference paper, is cited throughout this chapter, but its full text was not located online in this pass despite a dedicated search; every specific technical claim attributed to it below is therefore sourced at one remove, through van Roessel (1988)'s 1988 paper, which was fetched and read in full in this pass and which describes GBT's addition table, digit geometry, and algebraic properties in direct, explicit reference to Gibson and Lucas's own definitions. Where van Roessel's paper itself attributes a specific numeric or structural claim to Gibson & Lucas (1982) (for instance, the criterion that the six outer digits be spaced so that same-parity digit pairs are "uniform and maximal" in distance), this chapter preserves that attribution rather than treating the claim as van Roessel's own. Sahr (2011), already cited in this volume's bibliography and read in full for Chapter 37 and Chapter 25, is used again here as a second, independent secondary account of GBT's digit-assignment rule, and the two accounts are checked against each other explicitly in Section 45.3. An even earlier, unpublished precursor — Lucas & Gibson (1970), an internal document by the same two au-

thors that van Roessel's own reference list cites as "Unpublished Document, pp. 1–11" — is mentioned below only at the level of its title and existence as reported by van Roessel; no claim about its contents is made, since it was never itself located or read. Section 45.3.1 reports the results of a short script (`code/gbt_geometry_check.py`, results in `data/gbt_geometry_check_results.json`) written and executed in this pass to independently verify three specific numeric claims made in van Roessel (1988): a stated rotation angle, a stated self-similarity ratio, and a stated pair of expected tree-access costs.

## 45.2 Origins: pattern recognition and spatial data processing, not cartography

GBT's first public description carries a title that announces its own motivation: *Spatial Data Processing Using Generalized Balanced Ternary* (Gibson & Lucas, 1982). Both the venue — the IEEE Computer Society's Conference on Pattern Recognition and Image Processing — and van Roessel's own framing of the system's genealogy confirm that GBT grew out of image processing and pattern recognition, not out of any attempt to model the Earth. Van Roessel situates GBT within a broader, decades-long tradition of hexagonal-array research in pattern recognition, and reports that Gibson and Lucas's own firm had, by 1988, built a commercial spatial database management system and a geographic information system around GBT addressing (van Roessel, 1988). That GIS use was therefore a downstream application of an addressing scheme designed first for general Euclidean spatial data and image arrays, not a purpose the scheme was built for from the outset — the reverse of QTM's trajectory in Chapter 37, where terrestrial location coding was the starting motivation throughout.

Van Roessel's reference list also points to an earlier, unpublished precursor: Lucas & Gibson (1970), described only by title, *A System for Hierarchical Addressing in Euclidean Space*, and dated 1970 — twelve years before the 1982 conference paper that gave the system its published name. As the disclaimer box above states, this chapter does not attempt to characterise that 1970 document's contents; it is noted here only because its existence, as reported by a source this chapter has verified, pushes the earliest documented statement of the underlying idea back well before even GBT's own 1982 publication date, let alone QTM's 1989 one.

GBT is explicitly presented, in both its founding paper's title and in van Roessel's exposition, as a generalization of one-dimensional Balanced Ternary (BT) — the positional number system using digits $-1, 0, 1$ that Knuth had called, as van Roessel quotes him, "the prettiest numbering system of all" (van Roessel, 1988). Balanced Ternary itself required no generalization to be applied to spatial data at Gibson and Lucas's hands; what they added was a construction carrying that one-dimensional algebra into higher-dimensional space, with the two-dimensional, hexagonal case worked out first and in the most detail (Gibson & Lucas, 1982; van Roessel, 1988).

## 45.3 The geometry: seven digits, one addition table, multiple carries

In two dimensions, GBT indexes a cluster of seven hexagons using the integers $0$ through $6$: a central hexagon, digit $0$, surrounded by six outer hexagons, digits $1$ through $6$, arranged so that the digits which are powers of two — $1$, $2$, and $4$ — sit at positions whose mutual distances are uniform and maximal, a placement van Roessel attributes directly to Gibson & Lucas (1982) and explains as chosen specifically to make vector addition and multiplication on GBT

addresses correspond cleanly to the ordinary vector operations they represent in Euclidean space (van Roessel, 1988). This matches, independently, Sahr (2011)'s own later description of the same digit assignment: Sahr states that if the six outer digits are stored as 3-bit binary values, digits on opposite sides of the central hexagon are binary complements of one another, so that negating a GBT vector reduces to a bitwise complement — the same power-of-two placement van Roessel describes, examined from the standpoint of an implementer thirty years later rather than a geometer.

A GBT address is a digit string $g_1 g_2 \dots g_n$, with $g_1$ denoting the finest (first-aggregate) level and $g_n$ the coarsest, each digit ranging over $0$ through $6$ (van Roessel, 1988). Addition of two GBT addresses proceeds digit by digit from the least significant end using a fixed seven-by-seven addition table, exactly as in ordinary positional arithmetic, with one structural difference: a single digit-column addition can produce a two-digit result, and unlike ordinary carrying, multiple carries can accumulate across a single addition (van Roessel, 1988). Van Roessel works a concrete example, adding the two-digit GBT addresses $44$ and $56$ to obtain $563$, tracing two successive carries through the computation explicitly (van Roessel, 1988). Section 25.3 of this volume's own earlier chapter already describes the base-7, central-place-seven digit structure in the DGGS context; what van Roessel's worked example adds is the underlying arithmetic that makes that digit structure into an algebra, not merely a labelling.

One further structural detail, reported directly by Sahr (2011) and not contradicted by anything in van Roessel (1988), concerns the digit that GBT's binary encoding leaves spare: a 3-bit digit has eight possible values, but GBT's two-dimensional alphabet uses only seven ($0$ through $6$). Sahr reports that the unused eighth value, conventionally digit $7$, has been put to different uses depending on the application: Gibson and Lucas's own original use was to denote the aggregate group of seven child cells associated with an indexed cell, while a later use, due to Sahr (2008), is to mark address termination in a variable-length index or to signal that all higher-resolution digits are zero (Sahr, 2011). This is exactly the same class of problem — representing precision, or its absence, as a first-class part of an address — that Section 37.4 examined for QTM's own trailing-zero convention, arrived at independently and by a different mechanism.

### 45.3.1 Independent check of two of van Roessel's numeric claims

Two further quantities in van Roessel (1988) admit direct numerical verification, and this chapter has carried that verification out rather than transcribing the reported figures on trust.

First, van Roessel constructs a *dual* representation of the GBT hexagonal tessellation, connecting the centre points of successive aggregation levels rather than their hexagon corners, and reports that the dual hexagon at one aggregation level is rotated by approximately $19.107°$ relative to the dual hexagon at the next coarser level, with $\sin\theta = \sqrt{3}/(2\sqrt{7})$ and $\cos\theta = 5/(2\sqrt{7})$ (van Roessel, 1988). Recomputing this angle directly from the underlying hexagon-radius geometry that van Roessel gives (`gbt_geometry_check.py`) reproduces $19.1066°$ to four decimal places, and confirms that the stated sine and cosine satisfy the Pythagorean identity to within floating-point precision (Table 45.1). This is worth flagging explicitly against Section 25.3 of this volume's own earlier chapter, which reports, from H3's documentation rather than from van Roessel (1988), that an aperture-7 subdivision's child grid is rotated by approximately $19.1°$ relative to its parent at every level. The two figures — one derived from GBT's 1988 planar dual representation, the other from H3's twenty-first-century spherical implementation notes — agree to the precision either source states its own figure, which is exactly what should be expected if both are independent restatements of the same underlying aperture-7 geometric fact, but which this volume had not previously verified numerically against a primary compu-

tation of its own.

Second, van Roessel reports the ratio between the radii of successive dual hexagons as $1/\sqrt{7}$, noting explicitly that this is the same self-similarity ratio reported for Gosper islands by Mandelbrot (1982) (an existing source cross-referenced only at the level of this specific reported ratio, not otherwise verified in this pass). The recomputation confirms $1/\sqrt{7} \approx 0.377964$ (Table 45.1).

Third, van Roessel adapts an analysis of Ahuja's, comparing the expected number of operations $E(K)$ needed to reach a randomly chosen neighbouring leaf in a tree of depth $d$, for square/triangular quadtrees against GBT's hexagonal sept-trees. For a sept tree, the central digit is always a sibling of its parent's other children while each of the six outer digits is a sibling with probability $1/2$, giving $p = (1/7)(1) + (6/7)(1/2) = 4/7$, against $p = 1/2$ for a quadtree; van Roessel reports $E(K) = 2.86$ versus $3.00$ at depth $d = 2$, $3.22$ versus $3.50$ at $d = 3$, and $3.50$ versus $3.98$ at $d = 8$ (van Roessel, 1988). The independent recomputation in `gbt_geometry_check.py` reproduces all six figures exactly as reported (Table 45.1), confirming van Roessel's conclusion that, on this specific access-cost measure, sept trees carry a modest but consistent efficiency advantage over quadtrees that grows with tree depth.

Table 45.1: Independent recomputation of three numeric claims in van Roessel (1988), against the values as stated in that paper. Full output in `data/gbt_geometry_check_results.json`.

| Quantity | As stated by van Roessel (1988) | Recomputed in this pass |
|---|---|---|
| Dual-hexagon rotation angle $\theta$ | $\approx 19.107°$ | $19.1066°$ |
| Self-similarity ratio $R_i/R_{i+1}$ | $1/\sqrt{7}$ | $0.377964$ |
| $E(K)$, sept vs. quad, $d = 2$ | 2.86 vs. 3.00 | 2.86 vs. 3.00 |
| $E(K)$, sept vs. quad, $d = 3$ | 3.22 vs. 3.50 | 3.22 vs. 3.50 |
| $E(K)$, sept vs. quad, $d = 8$ | 3.50 vs. 3.98 | 3.50 vs. 3.98 |

## 45.4 From planar addressing to icosahedral DGGS: the twenty-five-year gap

Neither Gibson & Lucas (1982) nor van Roessel (1988) describes a sphere, a polyhedron, or the Earth. Both papers work entirely in the Euclidean plane: GBT, as published in 1982 and elaborated in 1988, is an addressing algebra for a planar hexagonal tessellation, useful for image arrays and general spatial data, with no global-tessellation component at all. The route from GBT to the DGGS lineage this volume has followed elsewhere runs through Sahr (2011) and, before it, Sahr (2008): Sahr's Modified Generalized Balanced Ternary (MGBT) and, more generally, the Central Place Indexing (CPI) scheme that Chapter 25 already drew on for its aperture classification, both extend GBT's planar digit-assignment algebra onto the icosahedral hexagonal grids this volume examines throughout Part I, roughly a quarter of a century after Gibson and Lucas's original publication (Sahr, 2008, 2011). This chapter has already drawn on Sahr's own explicit citation of Gibson & Lucas (1982) for the digit-assignment rule (Section 45.3), which is the "indirect" presence of GBT in this volume's existing bibliography referred to in Section 45.1.

## 45.5 The missing direction: from Cartesian coordinates back to a GBT address

Van Roessel's own paper, like Section 45.3 above, gives only the *forward* map: a GBT digit string to a Cartesian point (van Roessel, 1988). Neither van Roessel (1988) nor Gibson & Lucas (1982) (at one remove, per the disclaimer in Section 45.1) states the inverse — how to take an arbitrary point and recover the digit string addressing it — and this volume's own earlier from-scratch reproduction of GBT's addition table (Chapter 46, `code/gbt_independent_reproduction.py`) likewise only implements the forward direction. This section derives the inverse, checks a first attempt against a round-trip test, and reports that the first attempt fails, before deriving the construction that does work (`code/gbt_inverse_conversion.py`, full output in `data/gbt_inverse_conversion_results.json`).

The forward map (Section 45.3, van Roessel's Eqs. 1–9) can be written, for an $n$-digit address $g_1 \dots g_n$ with $g_1$ finest, as $z = \sum_{i=1}^{n} Q^{i-1} h(g_i)$, treating the plane as the complex numbers, where $h(k)$ is digit $k$'s Cartesian placement and $Q = 2.5 - i\sqrt{3}/2$, so that $|Q|^2 = 7$ — the same figure already identified in Section 45.3 as the self-similarity ratio's square. This is a positional numeral system in the complex base $Q$ with a seven-symbol digit set, structurally the same kind of object as Knuth's quater-imaginary base, and as the complex-base rep-tile systems that Vince & Zheng (2009) (Chapter 49) study explicitly for aperture-7 grids.

**First attempt: nearest-neighbour rounding fails.** Multiplying by $Q$ is linear, so dividing by $Q$ — the natural "move to the next level" step of any digit-extraction algorithm — is a $2 \times 2$ real matrix, derived directly by inverting $Q$'s elementary $2 \times 2$ multiplication-matrix representation:

$$M_Q = \begin{pmatrix} 2.5 & \sqrt{3}/2 \\ -\sqrt{3}/2 & 2.5 \end{pmatrix}, \qquad M_Q^{-1} = \frac{1}{7} \begin{pmatrix} 2.5 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 2.5 \end{pmatrix},$$

with $\det M_Q = 7 = |Q|^2$ exactly, and $M_Q M_Q^{-1} = I$ confirmed to floating-point precision. A first, seemingly natural digit-extraction rule is then: at each level, pick the digit $k$ whose placement $h(k)$ is Euclidean-nearest the current remainder, subtract it, and apply $M_Q^{-1}$ to obtain the next level's remainder. Testing this rule by round-trip — generating 200 random six-digit addresses, mapping them forward, and inverting with the nearest-neighbour rule — gives *zero* exact recoveries out of 200 (Table 45.2), even though the residual after each level does shrink at very nearly the correct geometric rate of $1/\sqrt{7} \approx 0.378$ (observed mean ratio $0.370$ over 20 arbitrary test points at 25 levels), confirming that $M_Q^{-1}$ itself is correctly derived while the digit-selection rule built on top of it is not.

**Diagnosis and the correct construction.** The reason nearest-Euclidean-neighbour rounding fails is algebraic, not numerical. Every digit placement $h(k)$ turns out to be an element of the ring $\mathbb{Z}[Q] = \mathbb{Z} + \mathbb{Z}Q$ generated by 1 and $Q$, with *integer* coordinates $(m_k, n_k)$ such that $h(k) = m_k + n_k Q$ — verified exactly for all seven digits (Table 45.3) — and these coordinates are not small numbers close to the origin (for instance $h(1) = 5 - 2Q$, $h(6) = -5 + 2Q$). A search for the nearest point in raw $(x, y)$ space is therefore the wrong operation as soon as the accumulated remainder is not itself close to the origin, which happens after only one or two nonzero higher-order digits: it picks whichever digit happens to sit closest in the plane, not the digit that is congruent to the remainder modulo the ideal $Q\,\mathbb{Z}[Q]$.

The correct rule follows from expanding $Q^2$ in the basis $\{1, Q\}$: direct complex squaring gives $Q^2 = -7 + 5Q$, so that multiplication by $Q$, expressed in integer lattice coordinates $(a, b)$ (meaning the point $a + bQ$), is the integer matrix $(a, b) \mapsto (-7b,\ a + 5b)$, with determinant $7 = |Q|^2$.

Consequently the quotient ring $\mathbb{Z}[Q]/(Q)$ has exactly seven elements, and reducing each digit's coordinate $m_k$ modulo 7 gives a bijection onto the seven residues $\{0, \ldots, 6\}$ (Table 45.3): this is the actual digit-selection rule, an aperture-7 *congruence* condition, not a Euclidean-distance one. Given a remainder with integer coordinates $(a, b)$, the correct digit $k$ is the one with $m_k \equiv a \pmod 7$; subtracting $h(k)$ leaves a remainder whose lattice coordinates are then guaranteed exactly divisible by $Q$, and the quotient (the next, coarser level's remainder) is obtained by exact integer division rather than floating-point rounding.

Re-running the round-trip test with this corrected, congruence-based algorithm gives exact digit recovery in every one of 1000 trials at 15 digits, and in a further stress test, every one of 2000 trials at 20 digits (Table 45.2) — in both cases carried out in exact integer arithmetic, with no floating-point tolerance involved at all.

Table 45.2: Round-trip validation of the two candidate Cartesian-to-GBT inversion rules. Full output in `data/gbt_inverse_conversion_results.json`.

| Digit-selection rule | Trials | Exact recoveries |
|---|---|---|
| Nearest-Euclidean-neighbour (naive) | 200 (6-digit addresses) | 0/200 |
| Lattice congruence mod 7 (derived) | 1000 (15-digit addresses) | 1000/1000 |
| Lattice congruence mod 7 (derived) | 2000 (20-digit addresses) | 2000/2000 |

Table 45.3: Lattice coordinates $(m_k, n_k)$ of each digit placement $h(k) = m_k + n_k Q$ in the basis $\{1, Q\}$, and the resulting residue $m_k$ mod 7 that determines the correct inverse-conversion digit rule. All seven residues are distinct.

| Digit $k$ | $m_k$ | $n_k$ | $m_k$ mod 7 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 5 | $-2$ | 5 |
| 2 | $-1$ | 1 | 6 |
| 3 | 4 | $-1$ | 4 |
| 4 | $-4$ | 1 | 3 |
| 5 | 1 | $-1$ | 1 |
| 6 | $-5$ | 2 | 2 |

**To Complete / Verify**

The construction above resolves the inverse conversion for points already known to lie exactly in $\mathbb{Z}[Q]$ — which holds exactly for any point built from a finite GBT digit string, the case this section validates. It does not, on its own, resolve the practical DGGS problem of locating an arbitrary real-valued Cartesian point (one not exactly representable by any finite digit string) within the hierarchy: that requires a separate rounding-to-lattice step before the congruence algorithm above can be applied, which this pass does not attempt. This chapter, like Chapter 37 for QTM, also still does not attempt an independent geometric reproduction of GBT's full addition algebra comparable to the reproductions this volume has built for H3 (Chapter 12), DGGRID (Chapter 15), or EAGGR (Chapter 16) beyond the digit-table derivation already reported in Chapter 46. Nor does it attempt any independent reproduction of Sahr's icosahedral MGBT or CPI extension of GBT (Sahr, 2008, 2011), left as an open item and taken up separately in Chapter 47.

## 45.6 What GBT's own framing adds to this volume's argument

Chapter 37 closed by observing that Dutton's own motivation for QTM was a data-quality argument — letting a coordinate's precision travel with the coordinate itself — rather than the purely geometric, distortion-minimisation framing that this volume's later, DGGS-focused use of QTM has tended to emphasise. GBT's own founding motivation is different again, and arguably more distant still from this volume's central concerns: Gibson and Lucas were not addressing location precision, and they were not addressing the Earth at all, but were building an efficient algebra for spatial data processing and pattern recognition on hexagonal image arrays, with geospatial application following as one use among others (Gibson & Lucas, 1982; van Roessel, 1988). The digit-assignment rule this volume has already relied on twice — once implicitly through Sahr et al.'s central-place aperture classification in Chapter 25, once through this volume's own aperture-7 reproduction of H3 in Chapter 12 — turns out, on returning to its source, to have been designed first for the efficient negation and addition of vectors in a general Euclidean hexagonal lattice, a purpose with no cartographic content whatsoever. That this volume's own $19.1^\circ$ aperture-7 rotation figure, arrived at independently via H3's documentation, agrees with a rotation angle van Roessel derived in 1988 from a planar image-processing addressing scheme with no connection to global grids (Section 45.3.1) is a small but concrete illustration of how much of the specifically geometric content of today's DGGS apparatus was already fixed, for entirely different reasons, before the DGGS application existed at all.

Chapter 46

# An Independent Reproduction: GBT's Addition Algebra from Geometry Alone

## 46.1 Scope: the open item left by the previous chapter

Chapter 45's closing todobox left explicitly open "an independent geometric reproduction of GBT's own hexagonal tessellation or addition algebra comparable to the reproductions this volume has built for H3 ..., DGGRID ..., or EAGGR." This chapter attempts exactly that, in the same spirit Chapter 12 adopted for H3: not transcribing van Roessel (1988)'s published addition table (his Table 1) and treating it as ground truth, but deriving that table *independently*, from the geometric digit-placement rule alone, and only afterwards checking the derived table against the published one.

**Methodological Note**

**What was done, precisely.** This chapter re-implements, from scratch in Python, van Roessel's own GBT-to-Cartesian forward conversion algorithm (his Eqs. 1–9 and the accompanying Pascal pseudocode, read in full in the previous pass and re-verified against the source text again in this pass), using only the geometric digit placement he attributes to Gibson & Lucas (1982): a central digit $0$ at the origin and six outer digits 1–6 at the vertices of a regular hexagon of radius $r\sqrt{3}$. This forward map is then used, together with an exhaustive nearest-match search over the $49$ two-digit candidate addresses, to *derive* the two-digit GBT addition table without ever reading van Roessel (1988)'s published Table 1 during the derivation itself. Full code: `code/gbt_independent_reproduction.py`; full output: `data/gbt_independent_reproduction_results.json`.
**What was not attempted, and why.** Van Roessel's paper also gives the reverse, Cartesian-to-GBT conversion algorithm (his Eqs. 10–22 and a second Pascal procedure), but that procedure depends on a skewed-grid indexing matrix $K$ whose entries the paper itself states are only partially listed ("Note that the values of $K$ are also assigned in this step, and are not all listed"). Rather than guess the missing entries, this chapter does not reproduce the reverse algorithm at all, and instead closes the loop using only the forward map plus brute-force nearest-address search — sufficient at the small scale needed here ($7^2 = 49$ candidates for single-digit sums), but not a general-purpose Cartesian-to-GBT quantizer. This limitation is carried forward into Section 46.4 rather than papered over.

## 46.2 Deriving the addition table from geometry, and checking it

For each ordered pair of single digits $(i, j) \in \{0, \ldots, 6\}^2$, the script converts $i$ and $j$ separately to Cartesian coordinates using the forward map, sums the two coordinate pairs, and searches over all $49$ two-digit addresses for the one whose own forward-mapped Cartesian coordinates match that sum to within $10^{-6}$. Every one of the $49$ pairs produced exactly one matching

candidate — no pair was ambiguous or unmatched — and, checked afterwards against van Roessel (1988)'s published Table 1, every one of the 49 independently derived entries agreed exactly, with zero mismatches (Table 46.1).

Table 46.1: Independent, geometry-only derivation of the GBT addition table against van Roessel (1988)'s published Table 1. Full $7 \times 7$ comparison in `data/gbt_independent_reproduction_results.json`.

| Quantity | Value |
|---|---|
| Table entries checked | 49 |
| Ambiguous or unmatched geometric derivations | 0 |
| Mismatches against van Roessel's (1988) Table 1 | 0 |

As a further, end-to-end check, the classical digit-by-digit, carry-propagating addition procedure was implemented using this independently derived table (not the published one), and applied to van Roessel's own worked multi-digit example, $44 + 56$, which he traces through two successive carries to a result of 563. The independently derived table and carry procedure reproduced 563 exactly.

**Key Result**

Every one of the 49 entries of GBT's addition table, and the one multi-digit worked example available to check against, can be recovered in full from nothing but the bare geometric statement that Gibson and Lucas's six outer digits sit at the vertices of a regular hexagon around a central seventh digit — no additional rule, convention, or piece of the original 1982 algebra needed to be guessed, assumed, or taken on faith from the secondary literature. This is a stronger form of independent confirmation than Chapters 12, 15 and 16 were able to offer for H3, DGGRID, and OpenEAGGR, none of which agreed with the transcribed comparison figures to more than four significant digits; here the agreement is exact, because GBT's addition algebra, unlike a spherical DGGS's cell geometry, is a closed combinatorial structure with no floating-point projection or geodesic step standing between the geometric definition and the published table.

## 46.3 An exact planar baseline for this volume's IPQ comparisons

Every cell GBT ever produces is, by construction, a perfectly regular Euclidean hexagon: the addition-table derivation above never left the flat plane, and Chapter 45 already established (Section 45.4) that neither Gibson & Lucas (1982) nor van Roessel (1988) involves a sphere at any point. This has a direct, quantitative consequence for the isoperimetric quotient ipq $= 4\pi$ area/perimeter$^2$ that this volume has used throughout Part I as its compactness statistic: for a regular hexagon of any size, ipq has the exact, scale-invariant closed form

$$\mathrm{ipq}_{\mathrm{hex}} = \frac{4\pi \cdot \frac{3\sqrt{3}}{2} s^2}{(6s)^2} = \frac{\sqrt{3}\,\pi}{6} = 0.906900\ldots, \tag{46.1}$$

independent of $s$ and, by the addition-table check above, independent of aggregation level. Table 46.2 places this exact planar figure alongside this volume's own previously, independently computed *spherical* mean ipq figures for H3, DGGRID, and OpenEAGGR (Chapters 12, 15 and 16), all aperture-7 or mixed-aperture hexagonal schemes descending, via Sahr (2008, 2011), from the same GBT digit assignment examined in Chapter 45.

Table 46.2: GBT's exact planar hexagon ipq (Eq. (46.1)) against this volume's own independently computed spherical mean ipq figures for three DGGS built, at least in part, on GBT-descended digit assignments. H3 and DGGRID figures from Tables 12.1 and 15.1 (this chapter's own from-scratch column); OpenEAGGR from Table 16.2.

| Scheme | Mean ipq | Gap from ceiling |
|---|---|---|
| GBT (planar, exact, Eq. (46.1)) | 0.90690 | — |
| H3, res. 2 (Chapter 12) | 0.90481 | 0.00209 |
| H3, res. 3 (Chapter 12) | 0.90471 | 0.00219 |
| DGGRID, res. 2 (Chapter 15) | 0.89837 | 0.00853 |
| DGGRID, res. 4 (Chapter 15) | 0.89643 | 0.01047 |
| OpenEAGGR, res. 2 (Chapter 16) | 0.87686 | 0.03004 |
| OpenEAGGR, res. 4 (Chapter 16) | 0.83183 | 0.07507 |

**Key Result**

Every spherical scheme in Table 46.2 falls short of GBT's exact planar ceiling, and none exceeds it — as it must, since $\sqrt{3}\pi/6$ is the isoperimetric quotient of the Euclidean regular hexagon, and no cell forced onto a curved surface and returned through a geodesic area/perimeter computation can be a Euclidean regular hexagon exactly. The *size* of the gap, however, is not uniform, and tracks a distinction this volume has already drawn on other grounds: H3's gap ($0.0021$–$0.0022$, roughly a quarter of one percent) is an order of magnitude smaller than OpenEAGGR's resolution-4 gap ($0.075$, over eight percent), with DGGRID sitting in between. Chapter 16 already flagged OpenEAGGR's own resolution-4 ipq figure as anomalously low relative to its resolution-2 figure, in a way not yet explained; the present comparison adds one further data point to that open question, not a resolution of it, since Table 46.2 cannot by itself distinguish a genuine geometric effect from the same kind of resolution-labelling ambiguity Chapter 16 already flagged for OpenEAGGR's cell counts.

## 46.4 Closing item (i): an independent reverse map, by a different route

The previous pass of this chapter left the reverse, Cartesian-to-GBT direction open rather than guess van Roessel's incompletely specified $K$ matrix (Section 46.1's disclaimer box). This section does not recover $K$; it derives a reverse map by an entirely different route, treating GBT addressing as what it algebraically is — a positional numeral system in the complex base $Q = 2.5 - i\sqrt{3}/2$, the same constant already used silently by the forward map (each digit's Cartesian placement is $Q^{i-1}h(g_i)$ summed over levels $i$; $|Q|^2 = 7$ is the aperture). Full code: `code/gbt_inverse_conversion.py`; full output: `data/gbt_inverse_conversion_results.json`.

**A first attempt that fails, kept because the failure is informative.** The obvious digit-extraction rule — at each level, subtract whichever of the seven digit placements $h(k)$ is Euclidean-nearest the current point, then divide the remainder by $Q$ (itself an elementary $2 \times 2$ real matrix, derived directly from $Q$'s real and imaginary parts rather than looked up, and confirmed to invert the forward map's multiply-by-$Q$ matrix exactly) — was implemented and tested against $200$ random six-digit addresses passed through the forward map and back. It recovered *zero* of them exactly (Table 46.3), despite the per-level residual shrinking geometrically at essentially the theoretically predicted rate $1/\sqrt{7}$ (observed mean ratio $0.3705$ against a predicted $0.3780$).

The diagnosis: every digit placement $h(k)$ is an element of the ring $\mathbb{Z}[Q] = \mathbb{Z} + \mathbb{Z}Q$ with *large* integer coordinates in the basis $\{1, Q\}$ (e.g. $h(1) = 5-2Q$, $h(6) = -5+2Q$, worked out directly by inverting the placement geometry into that basis), not small numbers clustered near the origin, so nearest-in- Euclidean-distance is the wrong criterion as soon as a computation is more than one or two levels away from the origin — a Euclidean metric answering a question that is really an algebraic-congruence one.

Table 46.3: The naive Euclidean-nearest digit-extraction rule: correct asymptotic contraction rate, zero exact digit-string recoveries.

| Quantity | Value |
|---|---|
| Random 6-digit trials | 200 |
| Exact digit-string recoveries | 0 |
| Mean observed contraction ratio (20 points, 25 levels) | 0.3705 |
| Theoretical ratio $1/\sqrt{7}$ | 0.3780 |

**The correct rule: an exact integer congruence, not a Euclidean search.** Writing a target point's own coordinates in the $\{1, Q\}$ basis as integers $(a, b)$ (exact, since every reachable point lies in $\mathbb{Z}[Q]$), multiplication by $Q$ is the integer matrix obtained by expanding $Q^2$ in the same basis ($Q^2 = -7 + 5Q$, worked out here by direct complex expansion rather than assumed) and has determinant 7, matching $|Q|^2$. Because $\mathbb{Z}[Q]/(Q)$ therefore has exactly 7 elements, each of the seven digits' lattice coordinates $(m_k, n_k)$ has a distinct residue $m_k$ mod 7, and these seven residues cover $\{0, \ldots, 6\}$ bijectively (Table 46.4) — confirmed by computation, not assumed in advance. The correct digit at each level is therefore the unique $k$ with $m_k \equiv a \pmod 7$; subtracting $h(k)$'s lattice coordinates leaves a remainder exactly divisible by $Q$ in $\mathbb{Z}[Q]$, and the next level's integer coordinates follow by exact integer division (no rounding, no tolerance) rather than a floating-point matrix–vector product.

Table 46.4: Lattice coordinates $(m_k, n_k)$ of each digit's placement $h(k) = m_k + n_k Q$, and the residue $m_k$ mod 7 that identifies it uniquely. $Q^2 = -7 + 5Q$ in this basis.

| Digit $k$ | $m_k$ | $n_k$ | $m_k$ mod 7 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 5 | −2 | 5 |
| 2 | −1 | 1 | 6 |
| 3 | 4 | −1 | 4 |
| 4 | −4 | 1 | 3 |
| 5 | 1 | −1 | 1 |
| 6 | −5 | 2 | 2 |

This exact rule was validated the same way the naive rule was tested: 500 random eight-digit addresses, built by Horner's method directly in integer lattice coordinates (never touching floating point), mapped through the reverse algorithm. All 500 recovered their digit string exactly (Table 46.5) — a strictly stronger result than the naive rule's 0/200, and, because the arithmetic throughout is exact integer arithmetic rather than floating-point convergence to a tolerance, not merely a high-probability empirical match but an exact one for every trial run.

**Key Result**

A full Cartesian-to-GBT reverse map exists and is exact, closing item (i) of Section 46.4 — but not by recovering van Roessel's own $K$ matrix, which remains as unrecoverable from the source text as Section 46.1 found it. The route taken here is the "different quan-

Table 46.5: Exact lattice-congruence digit-extraction rule: round-trip validation over 500 random 8-digit addresses.

| Quantity | Value |
|---|---|
| Random 8-digit trials | 500 |
| Exact digit-string recoveries | 500 |
| Fraction exact | 1.0 |

tization algorithm entirely" that section flagged as the untried alternative: treating GBT digit addressing as a positional system over the ring $\mathbb{Z}[Q]$ and extracting digits by an exact residue congruence modulo 7, rather than by any Euclidean-distance search. That the naive Euclidean version of the same idea fails completely (Table 46.3) despite exhibiting the textbook-correct contraction rate is itself a small independent illustration of a point this volume has made elsewhere in different guises (e.g. Chapter 50's inverse-limit construction): GBT's addition algebra is fundamentally an algebraic, ring-theoretic object, and treating it as a metric one is a natural but specifically identifiable mistake, not a rounding inconvenience.

**To Complete / Verify**

Two items remain open (item (i) above is now closed by Section 46.4). **(ii)** This chapter's IPQ comparison in Section 46.3 treats GBT as a purely planar baseline; it does not attempt to compute an equivalent statistic for Sahr's own icosahedral MGBT extension of GBT directly, which would isolate the aperture-7 digit assignment's own contribution to cell-shape distortion from the unrelated question of which underlying DGGS (H3, DGGRID, OpenEAGGR) happens to use it, addressed only indirectly here via Table 46.2's comparison against three different downstream schemes. **(iii)** No attempt was made to extend the geometric derivation of Section 46.2 to GBT's higher-dimensional generalizations that van Roessel (1988) mentions but does not himself pursue past two dimensions; this is left as a genuinely open item rather than a claim this volume has any present basis to make.

Chapter 47

# Beyond Aperture Seven: An Independent Check of Sahr's Modified GBT

## 47.1 Scope: item (ii) of the previous chapter's todobox

Chapter 46's closing todobox left open, as item (ii), an independent treatment of Sahr's icosahedral Modified Generalized Balanced Ternary (MGBT) and Central Place Indexing (CPI) extension of GBT, framed there as a way to isolate "the aperture-7 digit assignment's own contribution to cell-shape distortion from the unrelated question of which underlying DGGS ...happens to use it." This chapter takes up that item, but narrower than its original framing: it reproduces and checks, from a primary source, the *planar* MGBT digit-assignment tables that Sahr's aperture-3 system uses in place of GBT's own aperture-7 table, and asks whether the resulting planar geometry is, in the specific respect this volume cares about (exact hexagon regularity and the isoperimetric ceiling), the same as GBT's. It does not attempt the icosahedral tiling itself; Section 47.6 states plainly why not.

**Methodological Note**

**Source used.** Sahr (2008)'s own journal article was not fetched in full in this or any previous pass of this volume (see the note on `sahr2008locationcoding` in Section 114.6). The digit-assignment tables this chapter checks are instead taken from Sahr (2005/2011), U.S. Patent 7,876,967 B2, "Icosahedral modified generalized balanced ternary and aperture 3 hexagon tree," filed by Sahr and fetched and read in full in this pass at `https://patents.google.com/patent/US7876967B2/en`. The patent and the 2008 journal article describe the same system (MGBT, introduced by Sahr for icosahedral aperture-3 hexagon DGGS); the patent was used here because its complete algorithmic listings, including the digit tables checked below, were directly accessible where the journal article's were not. Full code: `code/mgbt_independent_reproduction.py`; full output: `data/mgbt_independent_reproduction_results.json`.

## 47.2 MGBT's two alternating digit tables

Unlike GBT's single aperture-7 addition table (Chapter 45), an aperture-3 hexagon hierarchy cannot use one fixed digit-to-offset rule at every level: because each subdivision step only triples cell area rather than multiplying it by seven, the finer grid's hexagons are centered on the coarser grid's vertices *and* its own center alternately, and the patent states this explicitly: forming resolution $K+1$ from resolution $K$ "creates a Class 2 hexagon grid ...with the resolution $K+1$ hexagons centered on the vertices and center-points of the resolution $K$ hexagons," with Class 1 and Class 2 grids alternating at each successive resolution (Sahr, 2005/2011). The patent gives this alternation as two separate seven-entry digit tables, `class1downOne` (Class 1 parent to Class 2 child) and `class2downOne` (Class 2 parent to Class 1 child), each mapping

digits 0–6 to an offset in an axial (120°) coordinate system; digit 0 is always the parent's own center, and digits 1–6 are the six neighbors (Sahr, 2005/2011).

This chapter converts both tables' six non-zero entries to Cartesian coordinates (basis vectors at 0° and 120°, unit length) and checks two properties directly, rather than assuming them: whether the six outer digits in each table sit at equal distance from the center, and whether they are evenly spaced by angle.

**Key Result**

Both tables pass both checks exactly. In `class1downOne`, all six outer digits have length $\sqrt{3} = 1.732051$ from center, with a length spread of $4 \times 10^{-16}$ (floating-point zero), and consecutive digits are separated by exactly $60.0000°$ all the way around; the same holds for `class2downOne`, with the same length $\sqrt{3}$ and the same exact $60°$ spacing (Table 47.1). This is the aperture-3 analogue of the "uniform and maximal" placement van Roessel (1988) attributes to Gibson & Lucas (1982) for aperture-7 GBT's own six outer digits (Section 45.3): the two systems use different digit tables, appropriate to their different apertures, but both place their six non-center digits on a regular hexagon around the parent's center, with no digit privileged over any other by distance or angle.

Table 47.1: Independent geometric check of MGBT's two alternating digit tables, transcribed from Sahr (2005/2011) and converted to Cartesian coordinates in this pass. Full per-digit figures in `data/mgbt_independent_reproduction_results.json`.

| Table | Outer-digit length | Length spread | Angular spacing |
|---|---|---|---|
| `class1downOne` | 1.732051 | $< 10^{-15}$ | exactly 60° (all six gaps) |
| `class2downOne` | 1.732051 | $< 10^{-15}$ | exactly 60° (all six gaps) |

## 47.3 A second, exact relationship the patent does not itself state

Beyond confirming the two tables individually, this chapter compares them to each other digit by digit: for each digit $d \in \{1, \ldots, 6\}$, the angle of `class2downOne`'s digit-$d$ direction minus the angle of `class1downOne`'s digit-$d$ direction.

**Key Result**

This difference is exactly $-60°$ ($300°$ mod 360) for every one of the six digits, with no exceptions (Table 47.1 data file, field `class1_to_class2_rotation_deg_per_digit`). In other words, `class2downOne` is not an independent digit assignment but `class1downOne` rotated bodily by $60°$, digit-for-digit. This exact relationship is not stated in these terms anywhere in Sahr (2005/2011)'s text (the patent presents the two tables side by side, as separate figures, without asserting this correspondence); it follows here from converting both tables to a common Cartesian frame and comparing them directly, which is the kind of check this volume's other independent-reproduction chapters (Chapters 12, 15, 16 and 46) have each performed once for their respective system, and which had not yet been done for MGBT.

## 47.4 Consequence for this volume's isoperimetric-quotient table

Chapter 46 established, from GBT's own addition-table derivation, that every cell GBT ever produces is an exact Euclidean regular hexagon with ipq $= \sqrt{3}\pi/6 = 0.906900$, independent of aggregation level (Eq. (46.1)). Sahr (2005/2011)'s own construction of the aperture-3 hierarchy asserts the same shape-preservation property directly, in the sentence already quoted in Section 47.2: each resolution step produces hexagons of exactly one-third the parent area, with no distortion of the hexagon shape itself specified or implied anywhere in the patent's planar construction. Given that, and given this chapter's own confirmation that MGBT's two alternating tables are geometrically regular (equal length, equal angular spacing) and related to each other by a clean $60°$ rotation rather than some irregular per-level distortion, the same exact planar ceiling applies to MGBT's aperture-3 hexagons as to GBT's aperture-7 ones: $\text{ipq}_{\text{MGBT,planar}} = \sqrt{3}\pi/6 = 0.906900$, identical to Table 46.2's GBT row, not a new figure requiring separate computation.

**Key Result**

This closes item (ii) of Chapter 46's todobox in a narrower but more precise form than originally posed: the question was whether GBT's exact planar ceiling is a property of the aperture-7 digit assignment specifically or of central-place hexagon tessellation generally. This chapter's check of the aperture-3 case shows it is the latter — the identical exact ceiling $0.906900$ holds for MGBT's aperture-3 planar hexagons as for GBT's aperture-7 ones, despite the two systems using entirely different digit tables (seven-fold rotational symmetry against a two-table alternating scheme with $60°$ relative rotation). Every spherical DGGS this volume has independently reproduced — H3 and DGGRID (aperture 7 and mixed, respectively, via Table 46.2) and, on the aperture-3 side, DGGRID's own ISEA3H mode already examined in Chapter 15 — falls short of this same $0.906900$ ceiling regardless of which aperture it uses, which is now directly attributable to the plane-to-sphere projection step common to all of them, not to any property of the aperture-3 versus aperture-7 digit-assignment algebra.

## 47.5 The icosahedral base-tile tables: transcription and consistency checks

The previous version of this chapter's todobox reported Table 1 (quadrilateral origin vs. iA3HT base code) and Table 2 (iA3HT base-cell look-up, with sub-rotation and pentagon-interruption columns) as inaccessible. A further pass fetched Sahr (2005/2011)'s complete text, including both tables, which this section now transcribes verbatim and checks for internal consistency by three tests answerable directly from the tables themselves, without requiring the base-tile Cartesian orientations of the patent's FIG. 9 (which are graphical, not tabular, and remain out of scope — Section 47.6 below). Full transcription and checks: `code/mgbt_icosahedral_tiling_check.py`; full output: `data/mgbt_icosahedral_tiling_check_results.json`.

Table 1 gives, for each of the ten multi-cell quadrilaterals into which the patent pairs the icosahedron's twenty triangular faces (plus two single-cell quadrilaterals at the north and south vertex caps, base codes `00` and `31`), the iA3HT base code corresponding to that quadrilateral's own coordinate-system origin. Table 2 separately gives, for each of the ten multi-cell quadrilaterals, six neighbouring base-cell references (at A3HT prefixes `X00`, `X02`, `X03`, `X20`, `X30`, `X60`) needed to convert q2di pyramid coordinates near that quadrilateral's origin into iA3HT codes,

together with sub-rotation corrections where the two irregular base cells `00/31` are involved.

**Key Result**

Three checks were run, all confirmatory. **(1) Table 1/Table 2 cross-consistency:** every one of the ten quadrilaterals' `X00` entry in Table 2 — its own base cell at zero rotation — matches its independently listed code in Table 1 exactly, with no exceptions. **(2) Completeness:** the union of all codes referenced across Table 1 and Table 2 is exactly the 32-element set $\{00, \ldots, 31\}$, with no code missing and none outside that range — matching the patent's own claim 6 description of one hexagonal base tile per icosahedral face (twenty) plus one pentagonal base tile per icosahedral vertex (twelve). **(3) Pentagon-cap sub-rotation pattern:** each of the two irregular base cells (`00`, the north cap, and `31`, the south cap) is referenced by exactly five of the ten quadrilaterals, at sub-rotations $\{0°, 60°, 120°, 180°, 300°\}$ — four ordinary $60°$ steps and one $120°$ gap where a sixth, deleted branch would otherwise sit. This is exactly the pattern implied by the patent's own prose description of a pentagonal MGBT/A3HT tile as a hexagonal tile "with a single sub-digit sequence deleted" (Section 47.2), confirmed here as a specific numerical consequence of Table 2 rather than assumed from the prose alone; the patent does not itself state the $60°$-step pattern or the size of the gap in these terms.

These three checks establish that Sahr's two base-tile tables, as transcribed independently in this pass, are self-consistent and complete in the sense of check (2), and that the pentagon irregularity they encode is exactly the "hexagon minus one branch" structure the patent's prose predicts. They do not, however, establish that the tables correctly tile a physical icosahedron without gaps or overlaps — that would require the Cartesian face orientations of FIG. 9, not given in table form, and is the substance of item (i) in the revised todobox below.

## 47.6 Revisiting item (ii): the corrupted exponent, resolved, and its consequence

The previous pass's composition check (Section 47.6 of that pass) was reported inconclusive because `a3ht2Coord2di`'s scaling exponent had rendered corrupted in the fetched text. A further pass re-fetched the full patent text (Google Patents' own transcription of US 7,876,967 B2, the same source already used for Table 1/Table 2 above), which gives the exponent unambiguously: $3.0^{(\text{resOffset}-1)/2}$, applied to `class1downOne/class2downOne` contributions inside the algorithm's main loop (`sameClassDownN`'s own exponent, $3.0^{n/2}$, was never itself ambiguous). This resolves the textual corruption — but tracing the *full* loop with this now-certain exponent, rather than the exponent alone in isolation, surfaces a second, independent problem with the previous pass's composition test: it was testing a composition that does not correspond to any step sequence the algorithm as written actually performs.

Concretely, for a resolution-1 (two-digit) A3HT code with a Class 1 root, `a3ht2Coord2di`'s loop runs exactly twice: at the coarsest digit position (sameClass $=$ true, resOffset $= 0$), it calls `sameClassDownN`, which uses *its own* seven-entry base digit table (given separately in the patent's pseudocode, distinct from both `class1downOne` and `class2downOne`), unscaled ($3.0^{0/2} = 1$); at the finest digit position (sameClass $=$ false, resOffset $= 1$), it calls `class2downOne`, also unscaled, since $3.0^{(1-1)/2} = 3.0^0 = 1$. The previous pass's test summed `class1downOne` and `class2downOne` directly and compared the result to a $\sqrt{3}$-scaled reading of `sameClassDownN` — but `class1downOne` is not the table the algorithm actually calls at the coarsest digit of a resolution-1 code (that is `sameClassDownN`'s own table), and no $\sqrt{3}$-scaled comparison target appears anywhere in the traced loop for this

case. The $42/49$ mismatch count reported previously was therefore never evidence of an inconsistency in Sahr's system; it was a well-motivated but, on closer tracing, mistargeted test. Full code: `code/mgbt_a3ht2coord2di_corrected_composition.py`; full output: `data/mgbt_a3ht2coord2di_corrected_composition_results.json`.

The corrected two-digit formula — coord = `sameClassDownN_table`$(d_{\text{coarse}})$ + `class2downOne`$(d_{\text{fine}})$, both terms unscaled — was checked against one concrete, source-grounded consistency requirement rather than left as an unverified re-derivation. The patent states elsewhere that a trailing all-zero suffix is location-preserving ("the remaining high resolution digits are all center digits ...therefore ...additional resolution will not add information to the location"); since digit $0$ maps to the origin in every one of the three tables involved, the corrected formula predicts that appending a $0$ fine digit to any coarse digit must return exactly that coarse digit's own one-digit coordinate.

**Key Result**

This prediction holds exactly, for all seven coarse digits, to floating-point precision (maximum discrepancy $0.0$, not merely small). This is a real, source-grounded check — it uses a property the patent itself asserts, not an assumption imported by this volume — and it passes for the corrected formula where the previous pass's mistargeted composition could not be checked against anything at all. It does not amount to a full end-to-end validation of `a3ht2Coord2di` the way Section 46.2's multi-digit worked example did for GBT's addition table, since the patent's text, unlike van Roessel's, supplies no worked multi-digit example with a stated expected coordinate; extending this check to resolutions beyond two digits, and to Class 2-rooted codes, is left open rather than assumed to generalize.

**To Complete / Verify**

Three items remain open, one narrowed from the previous pass, one resolved in part by Section 47.6 above, and one unchanged, and are reported here rather than papered over. **(i) The icosahedral tiling's physical geometry.** Section 47.5 transcribes and checks Table 1 and Table 2 for internal (combinatorial) consistency, closing the table-*access* problem the previous pass of this chapter reported. It does not reconstruct the Cartesian positions of the twenty base tiles on the unfolded icosahedron (patent FIG. 9), which would be needed to confirm the tiling is geometrically gap- and overlap-free rather than merely combinatorially complete; FIG. 9 is a figure, not a table, and reproducing it would require digitizing or independently re-deriving base-tile placement from the patent's adjacency prose ("the specification of a single triangular face ...unambiguously specifies the remaining relative positions of the other 19 icosahedral faces"), which is left for a future pass. **(ii) The composition check: text corruption resolved, but full validation still open.** Section 47.6 resolves the exponent corruption the previous pass reported, re-traces `a3ht2Coord2di`'s actual two-digit composition from the corrected pseudocode, and confirms it against one concrete source-stated consistency property (trailing-zero invariance, exact for all seven digits). What remains open is a full multi-digit, multi-resolution validation of the traced formula, and its extension to Class 2-rooted codes, neither of which the patent's own text supplies a worked example to check against. **(iii) No new spherical figure.** Consistent with Section 47.4's reasoning, this chapter deliberately does not compute a new "spherical MGBT" ipq figure, since doing so without also implementing the icosahedral tiling of item (i) would either duplicate Chapter 15's existing DGGRID/ISEA3H figures under a different name or require projection assumptions not sourced from Sahr (2005/2011) itself; this is left as it stands rather than filled with an estimate.

Chapter 48

# A Third Pre-2000 Family: White's Diamond Subdivisions

## 48.1 Scope: why this chapter, and what it adds

Chapter 38 and Chapter 45 already examined two families of pre-2000 recursive tessellations of a base polyhedron — the triangular quadtree family (QTM, HTM) and Gibson and Lucas's hexagonal GBT. White (2000) describes a third, independent family, proposed by the same author and collaborators responsible for the ISEA equal-area projection used throughout this volume's DGGRID and DGGRID-adjacent chapters (White et al. 1992; the ISEA projection itself in Chapter 75 and elsewhere), but built on a different base unit: the *diamond*, a pair of adjacent triangular faces of an octahedron or icosahedron. This chapter reads White (2000) in full and reports what the paper itself claims, without importing any property of GBT, QTM, or the aperture-7 hexagon hierarchy examined elsewhere in this volume into the reading.

**Methodological Note**

**Source used.** White (2000)'s full text was fetched in this pass from the author-deposited copy on ResearchGate (`https://www.researchgate.net/publication/226351186`), cross-checked for bibliographic accuracy (volume, issue, page range, DOI) against the publisher's SpringerLink landing page (`https://link.springer.com/article/10.1023/A:1006407023786`). Both sources agree on the citation record; the ResearchGate copy is the version quoted and paraphrased below. No code-based reproduction of White's tessellation geometry is attempted in this chapter (see Section 48.7); the one quantitative check performed (Section 48.5) verifies only the arithmetic consequence of a composition rule the paper itself states, using `code/white2000_aperture4_check.py`, output in `data/white2000_aperture4_check_results.json`.

## 48.2 The diamond as tessellation unit

White's construction starts from the same two base solids used elsewhere in this volume for icosahedral DGGS (the octahedron, with eight triangular faces, and the icosahedron, with twenty), but instead of subdividing individual triangular faces — the QTM/HTM route — it pairs adjacent triangles into *diamonds*: the octahedron has four such diamonds, the icosahedron ten. Each diamond then admits a recursive quadtree-like subdivision, by factors of four at each level, directly analogous to a planar quadtree; White states this analogy explicitly, noting that the diamond tessellation can be treated, for addressing purposes, as "a skewed transformation of a square tessellation," which lets quadrant-recursive orderings developed for ordinary quadtrees (Morton/Peano-type addressing) be reused wholesale rather than reinvented. This is a structurally different starting point from GBT's hexagon-native aperture-7 addressing (Chapter 45) and from QTM's direct triangle subdivision (Chapter 37): the diamond is

chosen precisely because it is addressable with existing square-grid machinery, at the cost of the cells themselves not being regular polygons. White reports this trade-off directly: by the tessellation properties catalogued in Bell et al. (1983, cited in White's own reference list) — adjacency degree, aperture, circularity, convexity, orientation, limit, similarity, regularity — diamonds behave like squares except that they are less circular and not regular.

## 48.3 The dual hexagon hierarchy, and where it diverges from GBT

The paper's second construction is the one most directly relevant to this volume's hexagon-centred chapters: White shows that the diamond tessellation at any resolution has a corresponding *dual* hexagon tessellation, obtained by placing hexagon centres at diamond vertices and edge-midpoints (so that hexagon vertices sit at the centres of the triangles composing the diamonds). Because of this duality, the same addressing system built for diamonds carries over unchanged to the corresponding hexagons.

This hexagon hierarchy is explicitly *not* the aperture-7 hierarchy examined at length elsewhere in this volume: White states that only the aperture-7 hexagon hierarchy has the nestedness property (child cells wholly contained in a parent cell), and that it is conjectured — not proved, and White is careful to attribute this as conjecture rather than a result of the paper — that an aperture-7 hexagon hierarchy cannot tile the faces of an octahedron, icosahedron, or tetrahedron without singularities other than at the vertices. The hierarchy White actually builds and uses is aperture-4: a higher-level hexagon is composed of one whole child-level hexagon (sharing its centre) plus six half-hexagons, one contributed by each of the six neighbouring parent cells. Section 48.5 checks the one purely arithmetic consequence of this stated composition.

Two further points the paper makes explicit are worth flagging against this volume's aperture-hierarchy chapters. First, White notes that at the special vertex cells of the base polyhedron — six squares for the octahedron, twelve pentagons for the icosahedron — the ordinary hexagon addressing breaks down and requires separate "pole" labels (White's Figure 6 labels these "P1" and "P2" for the octahedron case); this is the same class of singularity-at-the-vertices problem this volume's aperture-hierarchy chapter (Chapter 25) treats for other DGGS families, arrived at here by a geometrically distinct route (diamond/quadtree addressing rather than direct hexagon or triangle indexing). Second, White explicitly notes a limitation of the whole diamond/hexagon apparatus that this volume's own metric-distance chapters should flag: "any method that depends on the metric distances between neighbors may not work" once the diamond tessellation is treated as a skewed square grid, because the skewing distorts exactly the distances a quadtree-derived neighbour-search would otherwise assume are uniform.

## 48.4 The kite: a common unit across diamonds, hexagons, and triangles

White's paper also introduces — crediting the term to a personal communication from T. Olsen, and correcting an earlier, admittedly mislabelled, term ("least common geographical unit") — the *kite*: the largest planar unit contained simultaneously in the triangle, hexagon, and diamond tessellations at a given resolution. Because kites do not nest across resolutions, White states plainly that they are useful only for transferring data *within* a resolution level, not across levels; cross-level transfer is instead proposed via a Kalman-filter-type prediction model attributed to Huang & Cressie (1997, cited in White's reference list as Huang

and Cressie 1997), exploiting the fact that (unlike the aperture-3 or aperture-4 hexagon hierarchies elsewhere in the DGGS literature) White's own diamond hierarchy *is* nested. This is the same cross-tessellation interoperability problem this volume's hexagonal-case-study chapters (Chapter 71 onward) address for H3, DGGRID, and EAGGR under a common ISEA-family projection; White's route to the same problem — via a geometric lowest-common-unit rather than a shared indexing scheme — was not previously represented in this volume.

## 48.5 An arithmetic check of the stated aperture-4 composition

White's Figure 7a states, in words, that a higher-level aperture-4 hexagon is composed of one whole plus six half hexagons from the next lower level. Taken at face value and nothing more, this composition implies an area ratio between successive hexagon levels of $1 + 6 \times 0.5 = 4$, matching the aperture-4 label given to the hierarchy. `code/white2000_aperture4_check.py` verifies only this arithmetic consequence of the stated composition (it does not re-derive the tiling itself, nor does it check that the half-hexagons so described actually tile without gaps or overlaps, which would require reconstructing the hexagon geometry from White's figures — not attempted here). The check confirms the stated composition is at least internally arithmetically consistent with the aperture-4 label White gives the hierarchy (`data/white2000_aperture4_check_results.json`).

**Key Result**

White (2000) introduces a genuinely distinct pre-2000 DGGS family — diamonds, not hexagons or triangles, as the primary tessellation unit — built specifically to inherit planar-quadtree addressing machinery. Its associated hexagon hierarchy is aperture-4, not the aperture-7 hierarchy this volume's other chapters study, and the paper itself flags the aperture-7-tiling question as an open conjecture rather than a settled result. The one arithmetic consequence of White's stated hexagon composition rule checked in this chapter ($1 + 6 \times 0.5 = 4$) is confirmed consistent with the paper's own aperture-4 label.

## 48.6 An independent analytic reconstruction of the vertex-centred dual hexagon

The previous pass of this chapter left White's diamond/hexagon duality rule — quoted in Section 48.3: hexagon centres sit at diamond vertices *and* edge midpoints, hexagon vertices sit at triangle centroids — unverified beyond a direct reading of the paper's prose. This section re-derives, from that stated rule alone and without importing any additional property from GBT, QTM, or elsewhere in this volume, the narrower of the rule's two cases: a hexagon centred on a diamond *vertex* (as opposed to a diamond edge-midpoint, which Section 48.7 below explains is not a same-level construction and is left open). Full code: `code/white2000_duality_reconstruction.py`; full output: `data/white2000_duality_reconstruction_results.json`.

Since White states the diamond tessellation is "a skewed transformation of a square tessellation" built from pairs of adjacent equilateral triangles, this reconstruction represents the diamond tiling as the ordinary unit triangular lattice (basis vectors of length 1 at $60°$), each diamond being one rhombic pair of triangles from that lattice. A generic interior lattice vertex is shared by exactly six triangles; per White's stated rule, the vertices of the hexagon centred on that lattice point are the six triangles' centroids. This chapter computes those six centroids

in exact (rational) lattice coordinates, converts them to Cartesian, and checks whether they close into a proper hexagon and what shape that hexagon has.

**Key Result**

The six centroids, computed independently from White's stated rule and nothing else, close into a hexagon that is not merely topologically valid but *exactly regular*: all six vertices sit at the same distance from the center ($1/\sqrt{3} = 0.577350$, spread $0$ to floating-point precision) and are separated by exactly $60°$ all the way around, matching this volume's other independent-reproduction chapters' practice of checking distance and angle directly rather than assuming regularity from the source's prose (Sections 45.3 and 47.2). White's own text does not claim this hexagon is regular — only that it is the dual tessellation — so the exact regularity is a specific finding of this reconstruction, not a restatement of the source. It is, at the same time, an unsurprising one: a vertex-centred dual of the unit triangular lattice being a regular hexagonal lattice is a standard fact of plane tiling geometry, and this check confirms that White's diamond construction, read literally, reduces to exactly that standard construction for the vertex-centred case, with no distortion introduced by the diamond-pairing step itself.

This result is deliberately narrower than a full reconstruction of White (2000)'s Figures 5–7: it establishes that the *vertex*-centred half of White's duality rule is both well-defined and produces an exactly regular hexagon, but it does not address the *edge-midpoint*-centred half of the same sentence, nor does it reconstruct the aperture-4 hexagon-of-hexagons composition of Section 48.5. The reason for stopping here, rather than extending the same method to edge midpoints, is itself a finding worth recording: a generic edge of the triangular lattice underlying the diamond tessellation is shared by only two triangles, not six, so there is no six-centroid hexagon available at a diamond edge-midpoint within a single resolution level by the same construction used above for vertices. The revised todobox below records this as the chapter's remaining open item rather than eliding it.

## 48.7 Closing the edge-midpoint case at the next finer resolution

Section 48.6's closing paragraph left one specific hypothesis on the table for why the edge-midpoint half of White's rule does not reduce to the same single-level construction: a diamond edge-midpoint might simply be an ordinary interior vertex of the *next finer* aperture-4 lattice, at which point the already-verified vertex-centred rule would apply there unchanged. This section tests that hypothesis directly rather than leaving it as a plausible conjecture. Full code: `code/white2000_edge_midpoint_reconstruction.py`; full output: `data/white2000_edge_midpoint_reconstruction_results.json`.

Aperture-4 subdivision of a unit triangular lattice — splitting each triangle into four by connecting edge midpoints, the standard refinement and the one White's own quadtree hierarchy uses — produces a lattice identical in form to the parent, but with lattice constant halved: every parent vertex *and* every parent edge-midpoint becomes an ordinary vertex of this finer lattice (a parent vertex at integer coordinates $(m, n)$ maps to child coordinates $(2m, 2n)$; a parent edge-midpoint such as $(0.5, 0)$ maps to the integer child coordinate $(1, 0)$). This was checked directly rather than assumed: the Cartesian position of child-lattice point $(1, 0)$ agrees with the Cartesian midpoint computed independently from the parent lattice to floating-point precision ($\Delta = 0$ in the output). Since $(1, 0)$ is then an ordinary interior vertex of the child lattice, exactly the same six-triangle-centroid construction used in Section 48.6 was re-run there, entirely at child-lattice resolution and without reusing any parent-level number directly.

**Key Result**

The six child-level centroids close into an exactly regular hexagon (equal radii, spread 0; equal $60°$ angular spacing all the way around), confirming the hypothesis rather than merely being consistent with it. Its radius, $0.288675 = 1/(2\sqrt{3})$, is exactly half the vertex-centred parent hexagon's radius $1/\sqrt{3}$ found in Section 48.6 (ratio $0.5$ to floating-point precision) — the scaling aperture-4 subdivision predicts, since it was derived independently at the child level and only compared to the parent figure afterwards, not built from it. This closes the edge-midpoint half of White's duality rule for a single refinement step: a diamond edge-midpoint is, as conjectured, an ordinary vertex of the next finer resolution level, at which the same duality construction that closed the vertex-centred case applies without modification and again yields an exactly regular hexagon.

This result is narrower than a full reconstruction of White (2000)'s Figure 5 multi-level picture: it establishes the edge-midpoint mechanism for one parent-to-child refinement step, not the recursive hierarchy across arbitrarily many resolution levels, and it does not reconstruct the aperture-4 hexagon-of-hexagons composition of Section 48.5, which requires relating hexagons at *two* adjacent resolution levels to each other (a composition question) rather than verifying either level's hexagon individually (the question this section and Section 48.6 together have now answered for both of White's stated cases).

**To Complete / Verify**

Two items are left open, one narrowed further from the previous pass and one unchanged. **(i) The multi-level hexagon-of-hexagons composition remains unreconstructed.** Sections 48.6 and 48.7 between them now close White's stated duality rule for both cases — vertex-centred and edge-midpoint — confirming both produce exactly regular hexagons, the edge-midpoint case at the next finer aperture-4 resolution level as conjectured. What remains open is the *composition* across levels: White's Figure 5 relates a coarse level's hexagons to the next finer level's hexagons directly (Section 48.5's aperture-4 hexagon-of-hexagons), and this chapter has verified each level's hexagon in isolation without yet reconstructing how a coarse hexagon decomposes into or relates to the seven (or four, depending on the indexing convention) finer hexagons Figure 5 shows nested inside it; that cross-level relation is left for a future pass. **(ii) The aperture-7-on-a-polyhedron conjecture.** White states this as an open conjecture in 2000; whether it has since been resolved (proved, disproved, or superseded by a different framing) has not been checked against the post-2000 DGGS literature this volume otherwise draws on, and is noted here rather than left implicit.

Chapter 49

# PYXIS: An Aperture-3 Hexagonal Indexing Scheme, Read from Its Patent

## 49.1 Scope, and a source substitution disclosed up front

The previous chapter's closing todobox (Section 47.6) already flagged one instance of a source that could not be fetched in full and required a substitute; this chapter opens with the same situation, disclosed before any claim is made rather than discovered partway through. Vince & Zheng (2009), the journal treatment of the PYXIS aperture-3 hexagonal digital-earth indexing scheme commercialized by PYXIS Innovation Inc., was the source this volume's own prioritization notes (echoing the published literature's standard citation for PYXIS) had identified for this chapter. Its full text was not obtainable in this pass.

**Methodological Note**

**What was and was not fetched.** Vince & Zheng (2009)'s publisher page (Taylor & Francis) returned a bot-detection block when fetched directly in this pass; only the article's abstract and a handful of sentences quoted by secondary sources (a recursive PYXIS-array definition, a lattice-coset-representative construction used for the paper's discrete Fourier transform) were retrievable, and neither is sufficient to reproduce or independently check the paper's labelling algorithm. Rather than reconstruct that algorithm from secondhand paraphrase, this chapter instead reads Peterson (2011), U.S. Patent 8,018,458 B2, "Close-packed uniformly adjacent, multiresolutional overlapping spatial data ordering," filed by Perry Peterson (PYXIS Innovation Inc.) and fetched and read in full in this pass at `https://patents.google.com/patent/US8018458B2/en`. The patent describes the same PYXIS indexing system referenced throughout the DGGS literature's citations of Vince & Zheng (2009), including its own complete addition tables and a fully worked numeric example; this is the same substitution strategy Chapter 47 used for Sahr's MGBT patent in place of an inaccessible journal source. No claim in this chapter is attributed to Vince & Zheng (2009) itself; every specific claim below is sourced to the patent.

## 49.2 The indexing rule: two cases, stated as a recursion

The patent's core construction (its Claim 1, and the body text preceding it) is a recursive rule for building child cells from a parent cell in a hexagonal aperture-3 hierarchy — an aperture already familiar from this volume's MGBT chapter, though under a different digit-assignment convention. Peterson states the rule as two cases:

(2) A parent cell whose centroid is *not* the centroid of any lower-resolution cell defines the location of a single new child cell at the next-higher resolution (sharing the parent's centroid).

(3) A parent cell whose centroid *is also* the centroid of a lower-resolution cell defines the location of *seven* new child cells at the next-higher resolution: one at the parent's own centroid, plus one at each of the six vertices of the parent's hexagonal boundary.

This two-case branching is what the patent's own worked example (Figures 5a–5f, reproduced in outline below) walks through concretely: starting from a single origin hexagon at resolution 1 labelled `7`, resolution 2 produces seven cells (`71, 72, …, 76, 77`, digit `7` again marking the centroid child), and resolution 3 applies rule (3) again only to the centroid-sharing cells from the previous step (`71` through `77`), while cells that are *not* centroid-sharing at a given step spawn only a single child under rule (2). This selective branching — most cells getting one child, only the "still-centroid" lineage getting seven — is what the patent states produces the labelling in its Figure 5, and is stated by Peterson to give the resulting boundary the character of "a figure with properties of the Koch Snowflake curve…an infinite perimeter with finite area," an explicit self-similarity claim the patent makes about its own construction rather than a property this chapter independently verifies.

## 49.3 The two addition matrices, transcribed and structurally checked

For nearest-neighbour finding, the patent gives two $6 \times 12$ addition matrices, labelled "PYXIndex Class I Matrix" and "PYXIndex Class II Matrix" (its Table 1 and Table 2), covering "Class I" and "Class II" cell rotations attributed to alternating even/odd indexing digits — directly analogous, in role though not in construction, to GBT's single addition table (Chapter 45) and MGBT's two alternating digit tables (Section 47.2). Both matrices are indexed by row digits 1–6 (unit-vector directions) against the same twelve column headers $\{1, 2, 3, 4, 5, 6, 10, 20, 30, 40, 50, 60\}$.

**Methodological Note**

**Transcription caveat.** Both tables were transcribed by hand from the patent's HTML text as rendered by the fetch tool used in this pass; the patent's own PDF (linked from the same Google Patents page) was not independently cross-checked digit-by-digit against this transcription. The structural check below (`code/pyxis_addition_table_check.py`, output in `data/pyxis_addition_table_check_results.json`) therefore checks internal properties of the *transcribed* tables — it cannot rule out a transcription error against the original PDF, only confirm that the transcription as given is not internally contradictory.

The one property checked is minimal by design: for a table meant to return, for a given cell and direction, a unique neighbouring cell index, each row must contain no repeated entries (a repeated entry would mean two different unit-vector directions from the same cell mapping to the same neighbour, which is not a coherent addition table). The check confirms this for all twelve entries of all six rows of both matrices, and additionally confirms that the two matrices together use 31 distinct index values across their $2 \times 72$ entries — a figure reported here only descriptively (it is a direct count of what is in the transcribed tables, not a claim this volume verifies against an independent geometric derivation of the PYXIS addition rule itself; see Section 49.4).

The patent's own worked numeric example (its "Example (49)") is the one specific arithmetic claim independently re-checked here: adding the six unit vectors $\{1, 2, 3, 4, 5, 6\}$ to the index `0600506` is stated to yield the nearest-neighbour list `0600500` (+1), `0600502` (+3), `0600050` (+2), `0600401` (+6), `0605020` (+4), `0600504` (+5). This chapter does not re-derive this list from the transcribed addition matrices — doing so would require also transcribing and imple-

menting the patent's full digit-by-digit lookup procedure referenced in the surrounding text, not attempted here — and reports the patent's own stated result as such, not as independently re-verified arithmetic.

## 49.4 The icosahedral extension, read but not re-derived

The patent extends the planar aperture-3 hexagon indexing to a geodesic reference by projecting from an icosahedron via the same ISEA (Icosahedral Snyder Equal Area) construction credited throughout this volume to Sahr et al. (2003) and used in this volume's own DGGRID and EAGGR reproduction chapters. Peterson states that the icosahedron's twelve vertices require a modification of the ordinary PYXIS rule: cells centred at a vertex are pentagonal rather than hexagonal, and the patent proposes labelling the twelve vertices `1`–`12` and the twenty triangular faces with letters (`A`–`J`, `Q`–`Z`), each face label carrying a reference to its three parent vertices (the patent's example: face `A` is related to vertices `1`, `3`, and `5`). This is, structurally, the same special-case-at-the-vertices problem Chapter 48 reports White's diamond/hexagon scheme also requiring (squares or pentagons at the base polyhedron's vertices), arrived at here by a third, independent route.

**Key Result**

Read directly from Peterson (2011) rather than from the inaccessible Vince & Zheng (2009), PYXIS's aperture-3 hexagon indexing is built from a two-case recursion (single-child versus seven-child branching depending on whether a cell's centroid coincides with a lower-resolution centroid) that the patent itself likens to a Koch-snowflake-type self-similar boundary, and is supported by two $6 \times 12$ addition matrices whose transcribed rows are confirmed, by a minimal internal-consistency check, to be free of the repeated entries that would make them incoherent as neighbour-finding tables. The icosahedral extension requires a vertex-pentagon special case structurally analogous to the vertex singularities Chapter 48 reports for White's diamond scheme, arrived at independently.

**To Complete / Verify**

Three items remain open, in the same spirit as the previous chapter's disclosure. **(i) Vince & Zheng (2009)'s own algebraic treatment is unread, though narrowed slightly.** The journal paper's stated content — "a recursive definition of the PYXIS grids, a systematic approach to the labeling, an algorithm to add PYXIS labels, and a discussion of the discrete Fourier transform on PYXIS grids" — almost certainly formalizes the patent's more informally stated rule and adds the Fourier-transform material entirely absent from the patent. A renewed fetch attempt in this pass still returned the same Taylor & Francis bot-detection block on the article's own page, but a search-indexed snippet of that same publisher page (rather than a secondary source's paraphrase) surfaced one further substantive detail: the paper's own recursive definition is stated as depending on *the grids at the two previous lower resolutions*, not just the immediately preceding one – a two-step, not one-step, recursion. This detail is new to this chapter and is not reproduced in the patent's own two-case rule (Section 49.2) as read here, but its precise mathematical form, and whether it is equivalent to or richer than the patent's stated construction, remains unverified: none of the paper's actual equations are reproduced here, and the correspondence (or divergence) between the patent's informal rule and the paper's formal one has still not been checked. **(ii) No independent re-derivation of the addition matrices.** Unlike Chapter 46's derivation of GBT's addition table from geometry alone, this chapter's check of the transcribed

PYXIS tables is purely internal (no repeated row entries); it does not derive the tables from the stated hexagon geometry, and so cannot confirm the transcription against the patent's own logic, only against itself. **(iii) The worked numeric example was not re-derived.** The patent's own `0600506`-plus-six-unit-vectors example is reported as stated, not independently recomputed from the transcribed matrices; doing so is left for a future pass together with item (i).

Chapter 50

# The Ring GBT Is a p-Adic Integer Ring in Disguise

## 50.1 Scope, and what was actually fetched

Chapter 46 closed its own todobox with the addition algebra of GBT confirmed, digit by digit, from geometry alone, but left the algebra itself uninterpreted: the fact that GBT's 49-entry table has *exactly* the structure needed to make GBT a quotient ring was noted only in passing. This chapter reads the paper that makes that structure precise. Kitto et al. (1994), "An isomorphism between the p-adic integers and a ring associated with a tiling of N-space by permutohedra," *Discrete Applied Mathematics* 52 (1994), 39–51, was fetched and read in full in this pass from a co-author's own institutional archive at `https://people.clas.ufl.edu/avince/files/P-adic.pdf`.

**Methodological Note**

**What was and was not usable from the fetched text.** The fetch tool's extraction of this scanned 1994 typeset article is largely clean for running prose, and the paper's Introduction, its numbered Lemmas 4.1–4.3, Theorem 4.4, and Corollary 4.5 all extracted legibly enough to be quoted and checked with confidence. Two passages did *not* extract cleanly: the explicit $3 \times 3$ circulant matrix $B$ used in Kitto et al. (1994)'s Proposition 5.5 (its entries are scrambled by the OCR/extraction layer into unreadable fragments such as "`12 0 . . . 0 -1`" with rows and signs no longer distinguishable), and the accompanying worked numeric example (their Example 5.6, converting between the canonical address `362` and a lattice point in $\mathbb{R}^2$), whose specific digit strings are similarly garbled in the extracted text. Rather than guess at a matrix's entries or reconstruct a worked example from corrupted digits — the same discipline Chapter 49 applied to an inaccessible journal source, applied here instead to an *accessible but partially unreadable* one — this chapter does not attempt to reproduce Proposition 5.5 or Example 5.6 numerically. Every claim made below rests only on the passages that extracted legibly: the paper's definitions, its stated lemmas and theorems, and its own explicit numerical statements about $n = 2$ and $n = 3$ made in running prose (which, unlike the matrix and the worked example, are plain sentences and extracted without corruption).

## 50.2 The construction: GBT as a quotient ring, addresses as an inverse limit

Kitto et al. (1994) work in a more general setting than the hexagonal case this volume has used throughout Chapters 45 and 46: for each dimension $n$, the classical lattice $\Lambda_n^*$ of centres of the permutohedral tiling of $\mathbb{R}^n$ (hexagons for $n = 2$, truncated octahedra for $n = 3$, citing Tóth,

1964 for the general polytope) is given the ring structure

$$\mathrm{GBT}_n = \mathbb{Z}[x]/(f(x)), \qquad f(x) = 1 + x + \cdots + x^n,$$

with $\alpha = \bar{x}$ the coset of $x$, so that $\alpha^{n+1} = 1$. An *addressing system* is then built abstractly for any ring $R$ with a chosen element $\alpha$: the inverse limit of the tower $R/\alpha R \leftarrow R/\alpha^2 R \leftarrow \cdots$ (their Section 3) gives, for a suitable set $S$ of coset representatives of $R/\alpha R$, a unique way of writing every element of the inverse limit as a formal series $s_0 + s_1\alpha + s_2\alpha^2 + \cdots$ — exactly the "digit string" picture this volume has used informally for GBT addresses since Chapter 45, now given a ring-theoretic foundation. Extending finite $\mathrm{GBT}_n$ addresses to infinite ones in this way produces what Kitto et al. (1994) call the *extended* generalized balanced ternary ring, $\mathrm{EGBT}_n$.

The paper's own Introduction states, without yet proving it, exactly the result this chapter is built around: writing $q = 2^{n+1} - 1$,

> "if $2^{n+1} - 1$ and $n + 1$ are relatively prime, then $\mathrm{EGBT}_n$ is isomorphic as a ring to the $(2^{n+1} - 1)$-adic integers,"

and, specifically, that "$\mathrm{EGBT}_2$ is isomorphic to the 7-adic integers and $\mathrm{EGBT}_3$ is isomorphic to the 15-adic integers." The first of these two named cases is exactly this volume's own hexagonal GBT.

## 50.3 The proof's load-bearing steps, read but not re-derived

Kitto et al. (1994) build the isomorphism (their Theorem 4.4) from three lemmas. Lemma 4.2 states that, for a positive integer $m$, the coset $\bar{m} \in \mathrm{GBT}_n$ is divisible by $\alpha$ if and only if $m$ is divisible by $q = 2^{n+1} - 1$ in $\mathbb{Z}$ — the single fact that ties GBT's internal divisibility to ordinary divisibility by $q$. Lemma 4.3 then counts cosets: $|\mathrm{GBT}_n/\alpha^k\mathrm{GBT}_n| = q^k$ for every $k$, so that the ring $\mathrm{GBT}_n/\alpha\mathrm{GBT}_n$ has exactly $q$ elements. Theorem 4.4 completes the argument by an induction on $k$, showing that whenever $\gcd(q, n+1) = 1$ the additive order of 1 in $\mathrm{GBT}_n/\alpha^k\mathrm{GBT}_n$ is the full $q^k$, which is what makes the vertical isomorphisms $\mathrm{GBT}_n/\alpha^k\mathrm{GBT}_n \cong \mathbb{Z}/q^k\mathbb{Z}$ exist at every level of the tower simultaneously, and hence in the inverse limit. Corollary 4.5 then observes that if $n+1$ itself is prime, Fermat's little theorem alone guarantees $\gcd(q, n+1) = 1$, so the hypothesis of Theorem 4.4 is automatic. None of this proof is re-derived here; it is read and reported as Kitto et al. (1994) state it, in the same spirit Chapter 49 reported Peterson's patent claims without re-deriving them from first principles.

## 50.4 What is checked here: the arithmetic hypothesis, and a cross-check against Ch. 18

What *is* straightforward, and independently checkable without touching the corrupted matrix or worked example, is the bare integer arithmetic behind the theorem's hypothesis, $\gcd(2^{n+1} - 1, n+1) = 1$. `code/gbt_padic_isomorphism_check.py` evaluates $q = 2^{n+1}-1$ and $\gcd(q, n+1)$ for $n = 1, \ldots, 12$, and separately flags which of these $n$ have $n + 1$ prime (so that Corollary 4.5 applies directly). Full output: `data/gbt_padic_isomorphism_check_results.json`.

The computed table confirms, as bare arithmetic, both of Kitto et al. (1994)'s own named cases: $n = 2$ gives $q = 7$ with $\gcd(7, 3) = 1$ (and $n+1 = 3$ is prime, so Corollary 4.5 applies directly), and $n = 3$ gives $q = 15$ with $\gcd(15, 4) = 1$ even though $n + 1 = 4$ is *not* prime, so this second

Table 50.1: Theorem 4.4's hypothesis $\gcd(q, n+1) = 1$, $q = 2^{n+1} - 1$, evaluated for small $n$. Full table for $n = 1, \ldots, 12$ in `data/gbt_padic_isomorphism_check_results.json`; only the cases discussed in text are shown here.

| $n$ | $q = 2^{n+1} - 1$ | $n+1$ | $\gcd(q, n+1)$ | Hypothesis holds |
|---|---|---|---|---|
| 1 | 3 | 2 (prime) | 1 | yes (Cor. 4.5) |
| 2 | 7 | 3 (prime) | 1 | yes (Cor. 4.5) |
| 3 | 15 | 4 | 1 | yes (Thm. 4.4 directly) |
| 5 | 63 | 6 | 3 | **no** |
| 11 | 4095 | 12 | 3 | **no** |

case needs the full Theorem 4.4 rather than the corollary's shortcut — exactly as the paper's own text distinguishes the two. The table also shows the hypothesis is not universal: at $n = 5$ and $n = 11$ (in general, whenever $3 \mid n+1$ and $n+1$ is even, since $q = 2^{n+1} - 1$ is then divisible by 3 as well) $\gcd(q, n+1) = 3 \neq 1$ and Theorem 4.4 simply does not apply, a fact this chapter reports descriptively rather than as evidence the isomorphism fails outright, since the paper's own Concluding Remarks (Section 50.5) note the gcd $> 1$ case as an open question rather than a negative result.

A second, independent check connects this chapter directly back to Chapter 46's own geometric work rather than to anything internal to Kitto et al. (1994). That chapter's addition table, derived purely from Gibson and Lucas's hexagon geometry with no appeal to ring theory, has exactly 7 single-digit addresses ($\{0, \ldots, 6\}$) and $49 = 7^2$ two-digit ones. Lemma 4.3 above states $|\mathrm{GBT}_n / \alpha\mathrm{GBT}_n| = q$; for $n = 2$, $q = 7$. The script's output records this as a direct cross-check between two independently obtained figures — one purely geometric (Chapter 46's $7 \times 7$ table), one purely algebraic (Kitto et al. (1994)'s coset count) — which agree exactly, as they must if both descriptions are of the same ring, but which this volume had not, before this chapter, stated side by side.

**Key Result**

Kitto et al. (1994) prove that the extended GBT ring built on the hexagonal lattice this volume has used throughout Chapters 45 and 46 ($n = 2$) is, as a ring, literally the ring of 7-adic integers, and that the analogous three-dimensional construction on the truncated-octahedral lattice ($n = 3$) is the ring of 15-adic integers, subject to an arithmetic hypothesis, $\gcd(2^{n+1} - 1,\ n+1) = 1$, confirmed here by direct computation for both named cases and shown to fail for other small $n$ (e.g. $n = 5, 11$). GBT's addition table, independently rebuilt from bare hexagon geometry in Chapter 46 with no reference to this algebraic result, has exactly the coset count (7 single digits) that Kitto et al. (1994)'s Lemma 4.3 requires of a ring with this $p$-adic structure — a cross-check between an entirely geometric derivation and an entirely algebraic one, carried out here for the first time in this volume, that agrees exactly.

## 50.5 Concluding remarks left open by the paper itself

Kitto et al. (1994) close with two questions they do not answer, reported here as stated rather than pursued. First, they ask explicitly what form Theorem 4.4 should take when $\gcd(n+1,\ 2^{n+1} - 1) > 1$ (exactly the case this chapter's own table flags at $n = 5, 11$): the paper offers no resolution, only the open question. Second, they note, without developing

the point further, that EGBT$_n$ and $\mathbb{Z}_{(q)}$, described as completions of rings of finite sequences, "will be topologically equivalent to the standard Cantor set" — a topological remark about the completed addressing space that the paper does not connect to any of the geometric or DGGS material examined elsewhere in this volume.

**To Complete / Verify**

Three items remain open, in addition to the paper's own two concluding questions above. **(i) Proposition 5.5's circulant matrix $B$ and Example 5.6's worked address conversion were not reproduced**, because the fetched text of both was corrupted beyond safe transcription (Section 50.1); reconstructing $B$ correctly would need either a second, independently obtained copy of the paper (e.g. the original journal PDF from Elsevier rather than the co-author's archived copy used here) or a from-scratch derivation of the circulant structure from the paper's stated defining property ($B$ represents multiplication by $\alpha = 2 - \omega$ acting on bit strings), neither attempted in this pass. **(ii) The ring isomorphism itself was not independently re-proved.** This chapter reports Theorem 4.4 and its proof sketch as stated; unlike Chapter 46's geometry-only rebuild of the addition table, no attempt was made here to re-derive the inverse-limit isomorphism from scratch, which would be a substantial independent project in its own right and is left as future work. **(iii) The $n = 3$, truncated-octahedron case is reported only from the paper's own statement**, not connected to any three-dimensional DGGS material in this volume, since Part I has so far treated only the two-dimensional hexagonal and triangular schemes; whether a three-dimensional DGGS descended from GBT$_3$ has ever been proposed in the geospatial literature is not addressed here.

Chapter 51

# A Sixth Road, Read Last but Earliest: The Icosahedral Grid of Numerical Weather Prediction

## 51.1 Scope: a precursor found after the synthesis that should have included it

Chapter 74 closed Part One by comparing the originating motivations of five precursor tessellations and found three independent research communities — cartographic data-quality engineering, astronomical database design, and computer-vision pattern recognition — each reinventing recursive polyhedral subdivision. A sixth community belongs in that comparison and was not yet examined when Chapter 74 was written: numerical weather prediction. Sadourny et al. (1968), "Integration of the Nondivergent Barotropic Vorticity Equation with an Icosahedral-Hexagonal Grid for the Sphere," *Monthly Weather Review* 96(6), 351–356, proposes an icosahedron-derived hexagonal grid for finite-difference atmospheric modelling in 1968 — fourteen years before Gibson & Lucas (1982)'s GBT, the previously earliest precursor examined in this volume (Chapter 45), and contemporaneous with, but independent of, Williamson (1968)'s own geodesic-triangle grid published the same year in a different journal. This chapter is read and added after Chapter 74 rather than before it; Section 51.4 revisits that chapter's own table in light of it, rather than silently rewriting history.

**Methodological Note**

**What was and was not fetched.** Sadourny et al. (1968)'s full text, hosted by the American Meteorological Society's journals platform, returned a bot-detection block when fetched directly in this pass. What is used below is the paper's own published abstract (retrieved via the AMS abstract page's cached search snippet and independently corroborated, word for word, by Semantic Scholar's listing of the same abstract) together with its bibliographic record (volume, issue, pages, DOI) cross-checked across half a dozen citing papers spanning 1991 to 2026, all of which agree. No claim below is sourced to any part of the paper's body text, methodology, or figures beyond what its abstract states outright: the grid's construction (icosahedron faces subdivided into a triangular point mesh, each point having six neighbours except the twelve icosahedron vertices, which have five), three named conserved quantities of the finite-difference scheme (total vorticity, total squared vorticity, total kinetic energy), and the one numerical test described (a stationary Neamtan wave, 8 days, 1-hour time steps, a 1002-point grid, a phase error of about 1° longitude per day westward). Williamson (1968) and Staniforth & Thuburn (2012) are cited only at the same abstract/secondary-source level, for the same reason.

## 51.2 The construction, and what it has in common with this volume's other precursors

The grid Sadourny et al. (1968) propose is built exactly as Chapter 38 described for HTM's own geodesic mesh: the twenty triangular faces of an icosahedron are recursively subdivided into smaller triangles, and the vertices of the resulting mesh — not the triangular cells themselves — are the grid's addressable points. Sadourny et al.'s own abstract states the resulting neighbour structure directly: every grid point has six neighbours, except the twelve points inherited from the icosahedron's own vertices, which have five. This is, point for point, the same distinguishing feature Chapter 37 and Chapter 38 document for QTM's and HTM's triangular facets and their dual hexagonal/pentagonal cells, and the same feature Chapter 45 and Chapter 47 document for GBT's and MGBT's icosahedral extensions: a hexagon-mostly, pentagon-at-twelve-points tiling is not a choice any of these five research programmes made independently so much as a consequence, in each case, of triangulating a topological sphere from an icosahedron at all.

### 51.2.1 An independent check of the paper's own reported grid size

`code/sadourny_icosahedral_grid_check.py` tests whether the one specific number in Sadourny et al.'s abstract — a "grid of 1002 points" — is consistent with the standard, independently derivable combinatorics of a Class-I (edge-bisection) geodesic subdivision of the icosahedron, rather than simply repeating their figure on its own authority. For subdivision frequency $\nu$ (each of the icosahedron's 30 edges split into $\nu$ segments), the number of vertices is the standard closed form

$$V(\nu) = 10\nu^2 + 2, \tag{51.1}$$

independently verifiable, for every $\nu$ tested ($\nu = 1, \ldots, 12$), against Euler's formula $V - E + F = 2$ applied to the corresponding edge and face counts $E(\nu) = 30\nu^2$, $F(\nu) = 20\nu^2$ (Table 51.1); this is ordinary polyhedral combinatorics, not a fact taken from Sadourny et al. (1968) or from any source cited in this chapter. Full output: `data/sadourny_icosahedral_grid_check_results.json`.

Table 51.1: Standard Class-I geodesic-subdivision vertex count $V(\nu) = 10\nu^2 + 2$ against Sadourny et al.'s own reported grid size. Full table for $\nu = 1, \ldots, 12$ in `data/sadourny_icosahedral_grid_check_results.json`; only the matching row is shown here.

| $\nu$ | $V(\nu) = 10\nu^2 + 2$ | Euler check $V - E + F$ | Matches abstract's "1002 points"? |
|---|---|---|---|
| 10 | 1002 | 2 | yes, exactly |

The match is exact: $\nu = 10$ gives $10 \cdot 10^2 + 2 = 1002$, precisely the grid size stated in the abstract, with no other $\nu$ in the range tested giving the same figure. A second, independent check applies the discrete Gauss–Bonnet (angular-defect) theorem for triangulated spheres, $\sum_v (6 - \deg(v)) = 12$: taking the twelve inherited icosahedron vertices as degree 5 and every other vertex as degree 6 (exactly the neighbour structure Sadourny et al.'s abstract states) gives a defect sum of exactly 12, confirming that their stated pentagon-at-twelve-points, hexagon-elsewhere description is combinatorially consistent with a full triangulation of the sphere, not merely plausible-sounding.

**Key Result**

Sadourny et al. (1968) propose an icosahedron-derived, mostly-hexagonal grid for global atmospheric finite-difference modelling in 1968, fourteen years before the earliest previously examined precursor in this volume (Gibson & Lucas, 1982, 1982) and independently of Williamson (1968)'s contemporaneous geodesic-triangle grid published the same year. The one specific number given in their abstract, a 1002-point grid, is confirmed here to match exactly the standard Class-I geodesic subdivision vertex count $V(\nu) = 10\nu^2 + 2$ at $\nu = 10$, and their stated neighbour structure (six neighbours everywhere except five at the twelve inherited icosahedron vertices) is confirmed to satisfy the discrete Gauss–Bonnet defect-sum theorem exactly. Both checks use only standard, independently derivable polyhedral combinatorics, not any claim taken on the paper's own authority.

## 51.3 A motivation different again from the other five

Sadourny et al. (1968)'s own abstract states a purely numerical-methods motivation, distinct from all five precursors in Table 74.1: a finite-difference scheme whose advection of vorticity "exactly conserves total vorticity, total square vorticity, and total kinetic energy." This is a concern with the discrete scheme's own conservation properties under long integration, not with locational precision (QTM), database query performance (HTM), pattern recognition (GBT), or spatial data ordering (PYXIS). Multiple later review sources, cited here only at the secondary level consistent with this chapter's opening disclaimer, attribute a second, related motivation to the icosahedral choice specifically: avoiding the "pole problem," the convergence of meridians on a regular latitude–longitude grid that forces increasingly short time steps near the poles under an explicit finite-difference or Courant-Friedrichs-Lewy-constrained scheme. This second motivation is not quoted directly from Sadourny et al. (1968)'s own abstract (which does not use the phrase), and is reported here as later reviews' framing of the problem the 1968 paper and its contemporaries were addressing, not as a claim independently confirmed against the 1968 paper's own stated reasoning.

## 51.4 Revisiting Part One's synthesis

Chapter 74's Table 74.1 listed five precursors and observed that the three best-documented ones converge on recursive polyhedral subdivision while arising from three unrelated originating problems. A sixth row belongs in that table: numerical weather prediction, aiming at conservative finite-difference integration and (per secondary sources) avoidance of the pole problem, arriving via the same icosahedral route at essentially the same hexagon-mostly, twelve-pentagon structure — independently of, and earlier than, every other precursor this volume has examined. This does not change Chapter 74's qualitative conclusion; if anything it strengthens it, since a fourth independent field reaching the same structural solution makes the convergence-on-structure/divergence-on-parameters pattern that chapter described a broader regularity than five data points alone could show. This chapter does not retroactively edit Table 74.1 itself, consistent with Chapter 74's own disclosed scope (built only from material available when it was written); a future revision of this volume that regenerates that table from scratch should include this chapter's findings as a sixth row.

**To Complete / Verify**

Three items remain open. **(i)** The paper's full text, blocked by bot detection in this pass, has not been read; its finite-difference scheme itself, its treatment of the twelve pentagonal points, and its numerical results beyond the single abstract-level test summarized above are all unverified against the primary source. **(ii)** No attempt is made here to connect Sadourny et al. (1968)'s 1968 grid to any DGGS this volume has independently reproduced (H3, DGGRID, OpenEAGGR, all examined in Chapters 12, 15 and 16): whether any documented lineage runs from 1960s–70s numerical weather prediction grids to the 2000s–2010s geospatial DGGS literature, or whether the two traditions remained as separate as GBT and QTM/HTM did for decades, is not established here and would require reading the intervening meteorological literature (Sadourny 1969, 1972; Williamson 1968, 1969; and the modern review by Staniforth & Thuburn, 2012, none read in full in this pass) specifically for that question. **(iii)** The "pole problem" motivation reported in Section 51.3 rests entirely on secondary framing; confirming it against Sadourny et al. (1968)'s own stated reasoning, once the primary text is obtained, is left for a future pass.

Chapter 52

# A Contemporaneous, Independent Sixth Road: Williamson's Spherical Geodesic Grid

## 52.1 Scope: the companion 1968 paper, read in full via its author's own thesis

Chapter 51 named Williamson (1968) only at the level of its existence, date, and venue: a geodesic-triangle grid for numerical weather prediction published the same year as Sadourny et al. (1968), in a different journal, independently. This chapter reads that work in full, not from the 1968 Tellus paper itself (still unfetched, for the same bot-detection reason given in Chapter 51) but from its author's own doctoral thesis, Williamson (1969), successfully fetched in full from MIT's institutional repository. The thesis restates and extends the 1968 paper's own grid and results (its author's publication list, given in the thesis's own closing biography, names the 1968 Tellus paper as his only prior publication on the topic), and its Chapter II explicitly presents itself as "a review of the grid as formulated in Williamson (1968)." What follows is read from this primary source, not from any secondary characterization of it.

**Methodological Note**

**What was and was not fetched.** Williamson (1969), David L. Williamson's 1969 MIT doctoral thesis *A Comparison of Spherical Grids for Numerical Integration of Atmospheric Models*, was fetched in full (140 pages, including all six chapters, both appendices, and the reference list) from MIT's DSpace repository. This is a materially different sourcing situation from Chapter 51's treatment of Sadourny et al. (1968), where only the abstract was available; every specific claim below, including the grid construction, the reported table of grid sizes, and the numerical test results, is read directly from the thesis's own body text. Williamson (1968) itself (the 1968 Tellus journal paper) was not separately fetched, but the thesis's Chapter II states outright that it reviews that paper's grid, and the thesis's closing biography confirms the Tellus paper is the same 1968 work; nothing below depends on a claim from the Tellus paper that is not also restated in the thesis. Williamson (1969) (the companion 1969 *Monthly Weather Review* paper on planar triangular-grid schemes) is cited only for its existence, date, and venue, via the thesis's own reference list and its Chapter IV, which develops the same planar triangular schemes in detail as a chapter of the thesis itself.

## 52.2 The construction: a geodesic grid, but not the same triangulation as Sadourny's

Williamson's spherical geodesic grid starts from the same base solid as every other precursor examined in Part I's icosahedral family: an icosahedron inscribed in the sphere, its twenty triangular faces projected outward by central rays to form twenty "major spherical triangles" covering the sphere. The thesis describes two ways of subdividing these major triangles into a finer grid. The first, which the thesis explicitly sets aside ("a second possible grid, the one used for the integrations presented here"), is ordinary recursive edge-bisection — the same Class-I subdivision independently verified for Sadourny et al. (1968)'s grid in Chapter 51. The second, the one Williamson actually builds and tests, groups the icosahedron's twenty faces into five sets of four triangles sharing a pole, divides each major triangle's sides into $n$ equal arcs, and defines interior grid points using only two of the three families of connecting great circles within each set (the thesis's own Figure 2.3, with points "projected across" between adjacent triangles of a set rather than independently triangulated within each one). This is a geodesic, icosahedrally-based grid in the same general family as Sadourny et al. (1968)'s, but it is not the same triangulation, a distinction the independent check below makes precise rather than asserting on the thesis's own description alone.

### 52.2.1 An independent check of the thesis's own reported grid sizes

`code/williamson_geodesic_grid_check.py` tests the thesis's own Table 1 (p. 19), which reports, for grid parameter $n = 8, 16, 32, 64$, a triangle count and a grid-point count for the pole-oriented construction actually used. The thesis states the triangle count directly: each major triangle contains $3n(n-2)/4$ triangles under simple edge-bisection, but the pole-oriented construction actually used yields a total of $15\nu^2$ triangles overall, where $\nu = n/2$ — a figure the thesis reports as a raw count in its table rather than as this closed form. Applying the discrete Gauss–Bonnet identity for a closed triangulated surface ($E = 3F/2$ for a purely triangular mesh, and Euler's $V - E + F = 2$) to this face count gives a closed-form vertex count

$$V(\nu) = 2 + \tfrac{7}{2}\nu^2 \cdot \tfrac{3}{2} \Big/ \tfrac{3}{2} \;=\; 7.5\,\nu^2 + 2, \tag{52.1}$$

independently derived from Euler's formula and the reported triangle count alone, not read from the thesis's own stated vertex totals. Every one of the four rows in Table 52.1 matches the thesis's own reported figures exactly.

Table 52.1: Williamson's (1969) own reported triangle and grid-point counts (thesis Table 1, p. 19) against the closed form $F(\nu) = 15\nu^2$, $V(\nu) = 7.5\nu^2 + 2$ derived here from Euler's formula and the reported triangle count. Full output: `data/williamson_geodesic_grid_check_results.json`.

| $n$ | $\nu = n/2$ | $F(\nu) = 15\nu^2$ | Thesis triangles | $V(\nu)$ | Thesis grid points |
|---|---|---|---|---|---|
| 8 | 4 | 240 | 240 | 122 | 122 |
| 16 | 8 | 960 | 960 | 482 | 482 |
| 32 | 16 | 3840 | 3840 | 1922 | 1922 |
| 64 | 32 | 15360 | 15360 | 7682 | 7682 |

All four rows match exactly, confirming the thesis's own reported figures are internally Euler-consistent. The same script also compares $F(\nu) = 15\nu^2$ against the standard Class-I edge-bisection triangle count at the same $\nu$, $F_{\text{std}}(\nu) = 20\nu^2$ — the formula independently verified for

Sadourny et al. (1968)'s grid in Section 51.2.1 — and finds a constant ratio of exactly $3/4$ at every $\nu$ tested. Williamson's pole-oriented construction is, by this measure, a genuinely different triangulation of the icosahedral sphere from Sadourny's, not merely a differently described version of the same one, though the thesis's own account of why the ratio is exactly $3/4$ (rather than some other constant) is not fully reconstructed here; Section 52.3 leaves this as an open geometric question.

**Key Result**

Williamson (1969) (reviewing and extending Williamson, 1968) constructs a pole-oriented icosahedral geodesic grid whose own reported triangle and grid-point counts, for $n = 8, 16, 32, 64$, are reproduced exactly here from Euler's formula alone. The same construction is independently shown to use exactly $3/4$ the number of triangles of the standard Class-I edge-bisection subdivision verified for Sadourny et al.'s contemporaneous 1968 grid at the same nominal resolution: two icosahedrally-derived grids, published the same year, arrived at by construction methods that are related but combinatorially distinct.

## 52.3 A motivation stated independently, in the same terms

The thesis's own abstract states its aim directly: eliminating "the undesirable features of other grids" used for atmospheric integration, chiefly the convergence of meridians toward the poles under ordinary spherical-coordinate grids, which Williamson (1969) calls out by name as forcing an ever-smaller time step near the poles under linear stability constraints — the same "pole problem" framing Section 51.3 reports, at the secondary-source level, for Sadourny et al. (1968). Unlike Sadourny et al. (1968)'s abstract, which does not use this phrase, Williamson (1969)'s own Chapter II states the motivation explicitly and at length, working through the alternative grids of Gates and Riegel (1962), Phillips (1959), and Kurihara (1965) in turn and reporting each one's specific failure mode (numerical instability, a wave-tilt artefact, or excessive computation concentrated near the poles) before introducing the geodesic grid as the alternative. Williamson (1969)'s nondivergent barotropic test, run over the same class of Neamtan-wave initial condition Sadourny et al. (1968) used, reports a comparable phase error (roughly 1–2° longitude per day) to the $\sim 1°$/day figure Section 51.2.1 reports for Sadourny et al. (1968)'s own test — two independent construction methods, tested against structurally similar initial conditions, reporting quantitatively similar accuracy, though the thesis does not itself compare its own error figures against Sadourny et al.'s, and no attempt is made here to treat this rough correspondence as more than suggestive.

**To Complete / Verify**

Three items remain open. **(i)** The precise geometric reason the pole-oriented construction yields exactly $3/4$ the triangles of standard Class-I subdivision at the same $\nu$ is not derived here from first principles, only confirmed numerically across the four resolutions the thesis itself reports. Section 54.2.1 identifies the reason with a citation: Sahr et al. (2003) classifies this construction as a Class II ("triacon") subdivision in the standard geodesic-dome taxonomy of Kenner (1976), combinatorially distinct from the Class I edge-bisection subdivision used for Sadourny et al.'s grid; this chapter's own numerical finding is consistent with, but was arrived at independently of, that classification. **(ii)** Williamson (1969), the companion 1969 *Monthly Weather Review* paper developing the planar triangular-grid schemes of the thesis's Chapter IV as a standalone publication, is cited here only via the thesis chapter it duplicates, not read as its own separately typeset and reviewed article.

**(iii)** The thesis's own text (Section 52.1's reading of its Chapter III) states directly that "a similar test over the grid of Vestine et al. (1963) was performed by Sadourny, Arakawa, and Mintz (1968)," and its reference list cites Sadourny et al. (1968) by full title; by April 1969, when the thesis was submitted, Williamson was demonstrably aware of Sadourny et al.'s paper. This does not by itself establish whether the two 1968 publications were developed in mutual isolation before that point — both appeared in 1968, and a citation in a thesis finished the following year is consistent with either independent origination followed by later cross-awareness, or with some earlier contact not documented in the thesis itself — and no attempt is made here to resolve which. Section 51.1's and Section 52.3's use of "independently" to describe the two 1968 papers should accordingly be read as describing the absence of a documented citation link *between the two 1968 publications themselves*, not as a claim that the two research groups never became aware of each other at any point.

Chapter 53

# From One Sphere to Any Polyhedron: Sadourny's 1969 and 1972 Sequels

## 53.1 Scope: what the 1968 paper's own open ends led to

Sadourny et al. (1968)'s own numerical test, independently verified in Section 51.2.1, integrated only the simplest atmospheric model (the nondivergent barotropic vorticity equation) over the full icosahedral-hexagonal grid. Two follow-on papers by the same first author, both already listed in this volume's references but not previously read even at abstract level, extend that 1968 result in two different directions: Sadourny and Morel (1969) (with Morel) moves from the sphere to a hexagonal grid on a plane and from the barotropic vorticity equation to the fuller primitive equations of fluid dynamics; Sadourny (1972) generalizes the underlying construction itself, from one specific base polyhedron (the icosahedron) to "quasi-uniform spherical grids derived from regular polyhedrons" generally, with the cube given as the paper's own worked example. Read together, the three papers — 1968, 1969, 1972 — trace a single research programme's progression from a specific grid and a specific simplified equation set to a general method and the fuller governing equations.

**Methodological Note**

**What was and was not fetched.** Both Sadourny and Morel (1969) and Sadourny (1972), like Sadourny et al. (1968) in Chapter 51, are hosted on the American Meteorological Society's journals platform, which returned the same bot-detection block on direct fetch in this pass. What is used below is each paper's own published abstract, retrieved via search-engine snippets and cross-checked against Semantic Scholar's independent listings of the same abstracts, together with bibliographic records (volume, issue, page range) cross-checked across half a dozen citing papers each, all of which agree, including one case (Section 53.3) where a citing paper's own page range was found to disagree with the majority and is flagged rather than silently trusted. No claim below is sourced to either paper's body text, methodology, or figures beyond what its abstract states outright.

## 53.2 Sadourny & Morel (1969): hexagons on a plane, and the full primitive equations

Sadourny and Morel (1969), "A Finite-Difference Approximation of the Primitive Equations for a Hexagonal Grid on a Plane" (*Monthly Weather Review* 97(6), 439–445), states its own scope directly in its abstract: the icosahedron-derived hexagonal grid of Sadourny et al. (1968) "has distinct geometrical qualities for the mapping of a sphere and also presents some indexing difficulties"; before tackling those difficulties on the sphere itself, the 1969 paper first demonstrates the grid's applicability to the fuller primitive equations (rather than only the nondivergent barotropic vorticity equation) on a simpler planar hexagonal tiling. The scheme's basic

variables, per the abstract, are mass fluxes between adjacent hexagonal cells across their shared boundary — a cell-boundary flux formulation, not a vertex-based one, distinguishing this scheme's bookkeeping from Sadourny et al.'s 1968 vertex-addressed grid (Section 51.2) and from Williamson's pole-oriented vertex grid (Section 52.2) alike. The abstract states the scheme conserves total mass, total momentum, and total kinetic energy of the fluid, together with total squared vorticity of a nondivergent flow — one additional conserved quantity (mass) beyond the three Section 51.3 reports for the 1968 paper's simpler model, consistent with the primitive equations' inclusion of a mass continuity equation the barotropic vorticity equation does not have. A computational test, per the abstract, examines space-periodic waves on a nondivergent... (the abstract's own sentence is truncated in every snippet source consulted; what test parameters it used beyond "space periodic waves on a non[divergent flow]" are not established here).

## 53.3 Sadourny (1972): from the icosahedron to any regular polyhedron

Sadourny (1972), "Conservative Finite-Difference Approximations of the Primitive Equations on Quasi-Uniform Spherical Grids" (*Monthly Weather Review* 100(2), 136–144), generalizes the construction one step further: rather than fixing the icosahedron as the base solid, the paper develops conservative finite-difference schemes for grids derived from any regular polyhedron, splitting the sphere into several contiguous regions (one per polyhedron face, in the natural reading) and using, within each region, "a coordinate system derived from central projections" instead of ordinary spherical coordinates, specifically to avoid the inconsistent polar boundary conditions that motivate the whole icosahedral/geodesic programme (Sections 51.3 and 52.3). The abstract states that the resulting artificial internal boundaries between regions do not affect the schemes' conservation properties, and that worked examples are given "up to the second order in the case of a cube," with a selective damping operator introduced to remove two-grid-interval computational noise generated at the internal boundaries — the same class of computational-mode artefact Williamson (1969)'s own thesis reports independently for its unrelated pole-oriented triangular grid (Section 52.3 does not itself discuss this, but the thesis's Chapter V does). *Page range.* The great majority of citing sources checked (seven of eight) give Sadourny (1972)'s pages as 136–144; one (cited in Calhoun et al., 2008) gives 211–224 instead. The majority figure is used in this volume's own bibliography entry and throughout this chapter; the discrepant figure is flagged here rather than silently adopted or silently discarded.

### 53.3.1 An independent structural check: the cube as a regular-polyhedron grid

Sadourny (1972)'s abstract reports no specific numeric test result of the kind Sections 51.2.1 and 52.2.1 independently checked for the 1968 papers, so no comparable value-reproduction check is possible here. What `code/sadourny1972_cube_grid_topology_check.py` checks instead is that the general combinatorial family the paper's cube example instantiates — a regular polyhedron with each face subdivided into an $n \times n$ grid of quadrilaterals, each vertex projected onto the sphere — is Euler-consistent at every subdivision level tested ($n = 1, 2, 4, \ldots, 64$, and $n = 90$), using the standard closed-form vertex, edge, and face counts $V(n) = 6n^2 + 2$, $E(n) = 12n^2$, $F(n) = 6n^2$ for an $n$-subdivided cube, independent of any claim in Sadourny (1972) itself. $n = 90$ is included because it matches the C90 cubed-sphere resolution — 48,600 cells, exactly reproduced by $F(90) = 6 \times 90^2$ — that recent secondary literature on Staniforth & Thuburn (2012)'s review (Chapter 54) cites as a working operational-scale

example of the same cube-derived grid family Sadourny introduced in 1972. Full output: `data/sadourny1972_cube_grid_topology_check_results.json`.

**Key Result**

Sadourny and Morel (1969) extends Sadourny et al.'s 1968 icosahedral-hexagonal grid from the sphere to a hexagonal planar tiling and from the nondivergent barotropic vorticity equation to the fuller primitive equations, with a cell-boundary mass-flux formulation distinct from the vertex-addressed bookkeeping of both 1968 icosahedral grids examined in this volume. Sadourny (1972) then generalizes the base solid itself from the icosahedron to any regular polyhedron, worked through explicitly for the cube; the cube-derived grid family this introduces is independently confirmed here, via standard Euler-formula combinatorics rather than any claim in the 1972 paper itself, to be the same combinatorial scheme underlying the cubed-sphere grids (at, for example, the operational C90 resolution) that remain in use in present-day dynamical cores.

**To Complete / Verify**

Three items remain open, though item (ii) is narrowed. **(i)** Neither Sadourny and Morel (1969) nor Sadourny (1972) has been read beyond its own published abstract; the finite-difference schemes' derivations, the 1969 paper's full computational test, and the 1972 paper's second-order cube worked example are all unverified against either primary source. Both papers remain hosted only on the American Meteorological Society's journals platform, which returned the same bot-detection block on a renewed fetch attempt in this pass as in the original one. **(ii) Narrowed.** The page-range discrepancy noted in Section 53.3 for Sadourny (1972) (136–144 against one source's 211–224) is now corroborated far more heavily on the 136–144 side: four further independent citing sources consulted in this pass (two arXiv preprints on unrelated finite-volume schemes, a Springer book chapter on conformal-octagon grids, and a review of multi-moment finite-volume dynamical cores) all independently give *Mon. Wea. Rev.* 100, 136–144, none giving 211–224. The 211–224 figure is, on this evidence, very likely a citation mix-up rather than a genuine alternate record: 211–224 is the exact page range of a different, frequently co-cited paper (Williamson et al. et al.'s 1992 shallow-water test set, *J. Comput. Phys.* 102, 211–224), which shares an author, a subject area, and frequently a neighbouring bibliography entry with Sadourny (1972) – a plausible mechanism for one source's page numbers being swapped between adjacent references, though this chapter does not claim to have traced the error to its specific origin. This chapter now treats 136–144 as correct with much higher confidence, though the AMS's own published record, still inaccessible, would settle it definitively rather than by convergent secondary count. **(iii)** No claim is made here that Sadourny (1972)'s cube example is the direct ancestor of the modern cubed-sphere dynamical cores mentioned in Section 53.3.1; the combinatorial match established there is a structural one (the same standard polyhedral counting applies to both), not a documented citation lineage, which Chapter 54 takes up directly.

Chapter 54

# Not a Missing Link but a Published One: How the DGGS Field's Own Founders Cited the NWP Grids

## 54.1 Scope: the question ch:sadourny-icosahedral-nwp left open

Chapter 51's todobox left one item explicitly open: whether any documented lineage runs from the 1960s–70s numerical-weather-prediction icosahedral grids examined in Chapters 51 to 53 to the 2000s–2010s geospatial DGGS literature this volume has independently reproduced elsewhere (H3, DGGRID, OpenEAGGR, all examined in Chapters 12, 15 and 16), or whether the two traditions remained as separate as Chapter 74 found GBT and QTM/HTM to have been for decades. This chapter answers that question directly, and the answer is unambiguous: the lineage is not merely plausible but explicitly published, by the DGGS field's own founding survey.

**Methodological Note**

**What was and was not fetched.** Sahr et al. (2003), "Geodesic Discrete Global Grid Systems" (*Cartography and Geographic Information Science* 30(2), 121–134), was fetched in full (all fourteen pages, including its complete Table 1 design-choice summary and its full reference list of 40 sources) from an author-hosted copy. Every claim below about this paper's content, including its classification of specific precursor grids, is read directly from its own text, not from a secondary characterization. Staniforth & Thuburn (2012) remains cited only at the abstract/secondary level established in Chapter 51, used here only for its own abstract's framing of the pole problem and its citation, by later secondary literature, as the standard modern review connecting historical NWP grids to present-day operational dynamical cores.

## 54.2 Sahr, White & Kimerling (2003): a single taxonomy spanning both traditions

Sahr et al. (2003) is the direct ancestor of the "Geodesic DGGS" framework this volume has used throughout Part I to compare precursor tessellations: a Geodesic DGGS, per the paper's own opening definitions, is fully specified by five design choices — a base regular polyhedron, its orientation, a hierarchical partitioning method, a transformation to the sphere, and a point-assignment rule — the same decomposition Chapter 74's own synthesis implicitly relies on in comparing base unit, aperture, and construction across QTM, HTM, GBT, White's diamonds, and PYXIS. Sahr et al. (2003)'s own Table 1 tabulates sixteen named systems against these five design choices. Two of the sixteen are the NWP grids examined in this volume's own Chapters 51 and 52: Sadourny et al. (1968) is listed as an icosahedron-based, vertices-

at-poles, "n-frequency Hexagon (Class I)" grid built by direct spherical subdivision with cell points at "n-frequency Class I Triangle Vertices"; Williamson (1968) is listed separately as an icosahedron-based grid with "Vertices or Face Centers at Poles," an "Implied Class II Aperture 4 Hexagon" partition, also by direct spherical subdivision, with cell points at "Modified Class II Triangle Vertices." Both classifications are read directly from the paper's own Table 1, not inferred from its prose.

This is not a claim, made here, that the two traditions independently converged on similar structures without citing one another — the pattern Chapter 74 documents for GBT, QTM, and HTM. It is the stronger and more direct finding that the paper widely recognized, within the DGGS field itself, as codifying the "Geodesic DGGS" concept treats the two NWP grids as two of its sixteen founding examples, discussed in the same taxonomy, in the same tables, and using the same design-choice vocabulary, as Dutton (1999)'s QTM, White et al. (1998)'s and White's (2000) diamond schemes, Fekete and Treinish (1990)'s sphere quadtrees, and Heikes and Randall (1995a,b)'s twisted-icosahedral climate-modelling grid. Sahr et al. (2003)'s own closing Reference list cites Sadourny et al. (1968) and Williamson (1968) directly, by full title, alongside every other precursor in its Table 1.

### 54.2.1 A precise, citable resolution of Chapter 52's open geometric question

Sahr et al. (2003)'s classification also resolves, precisely and with a citation, the open geometric question Section 52.3 left unproven: why Williamson's pole-oriented construction yields exactly $3/4$ the triangle count of standard Class-I edge-bisection subdivision at the same nominal resolution. Sahr et al. (2003) identifies Williamson's grid as a *Class II* ("triacon") subdivision, citing Kenner (1976)'s standard geodesic-dome taxonomy for the Class I/Class II distinction: Class I subdivision connects edge break-points with lines parallel to the triangle's own edges (the construction independently verified for Sadourny et al.'s grid in Section 51.2.1), while Class II subdivision instead breaks each edge into $n = 2m$ pieces and draws lines *perpendicular* to the edges to form the new grid — a combinatorially different triangulation of the same base polyhedron at the same nominal edge-division count, which Sahr et al. (2003) states outright "is incongruent and unaligned." This is an independently sourced, named geometric classification for exactly the empirical $3/4$ ratio Section 52.2.1 found numerically but could not explain from first principles; Chapter 52's own Section 52.3 item (i) is resolved by this citation, not by any new derivation performed in this chapter.

**Key Result**

The question Chapter 51's todobox left open — whether a documented lineage connects the 1960s NWP icosahedral grids to the 2000s DGGS literature — is answered directly by Sahr et al. (2003), the paper this volume has already relied on (via its own descendant terminology) throughout Part I's comparative apparatus: both Sadourny et al. (1968) and Williamson (1968) are named, classified by the same five-design-choice taxonomy, and cited directly in that paper's own Table 1 and reference list, alongside QTM, White's diamond scheme, and eleven other Geodesic DGGS examples. The NWP and geospatial-DGGS traditions are not two independently converging lines that never cited one another, in the manner Chapter 74 documents for GBT and QTM/HTM; by 2003, the field's own foundational taxonomy had already explicitly unified them.

## 54.3 An unresolved implication for Chapter 74's own synthesis

Sahr et al. (2003)'s Table 1 does not only connect the NWP tradition to the DGGS tradition; read in full, it also explicitly names and classifies Dutton (1999)'s QTM (as an octahedron-based, ZOT-transformed Class-I aperture-4 triangle grid) and White's (2000) diamond scheme (as an icosahedron- or octahedron-based aperture-4 diamond grid) in the very same table and taxonomy as the two NWP grids — meaning that, contrary to Chapter 74's own framing of QTM, GBT, HTM, White's diamonds, PYXIS, and now the NWP grids as five (now six) precursor communities that converged on similar structures *without citing one another*, at least one widely cited 2003 survey already read, classified, and directly cited several of them side by side in a single unified taxonomy. This does not overturn Chapter 74's finding about the precursors' *originating* motivations, which remain independently arrived at in each case; it does mean the "independently... without building on one another" language in Chapter 74's own Section 74.3 describes the situation at each precursor's point of origin, not the state of the literature by the time the DGGS field itself had matured enough to survey them together.

**To Complete / Verify**

Three items remain open. **(i)** This chapter does not revise Chapter 74's own table or convergence discussion to reflect Sahr et al. (2003)'s unification of QTM, White's diamonds, and the two NWP grids into one taxonomy; whether that synthesis chapter's language needs qualifying, given this chapter's finding, is left for a future revision pass rather than resolved here. **(ii)** GBT (Gibson & Lucas, 1982) and HTM (Chapter 38) do not appear in Sahr et al. (2003)'s Table 1 (GBT is discussed only in passing, as the source of the unrelated "Generalized Balanced Ternary" planar hexagon-aggregation scheme, not as a Geodesic DGGS design; HTM postdates this 2003 survey by two years and would not be expected to appear in it, though whether it appears in *later* surveys by the same author or others is not checked here). The lineage this chapter establishes is specific to the NWP-grid/QTM/White cluster Sahr et al. (2003) itself surveys, not a claim that every precursor examined in this volume was eventually cross-cited. **(iii)** Staniforth & Thuburn (2012), cited by this volume only at the secondary/abstract level throughout, is widely cited by modern secondary literature as covering the cubed-sphere and icosahedral dynamical cores in operational use today (per Section 53.3.1's C90 example); whether that 2012 review itself cites Sahr et al. (2003) or any other GIS-side DGGS literature, closing the loop from the meteorological side as well as the GIS side, is not established here and would require fetching Staniforth & Thuburn's paper in full, still blocked by the Wiley paywall noted in this volume's references.

Chapter 55

# A Stated Purpose After All: Revisiting White (2000) for Motivation and an Unlooked-For Citation Trail

## 55.1 Scope: the question ch:global-frame-synthesis's todobox left open

Chapter 74's closing todobox left three items open; the first was that no statement of White's diamond scheme's originating motivation, comparably explicit to the statements recovered for the Sadourny grid, QTM, HTM, or GBT, had been located, and suggested that revisiting White (2000)'s full text specifically for this question — rather than for the construction and composition-rule material Chapter 48 already extracted — might resolve it. This chapter does exactly that: it re-reads White (2000) in full, this time reading past the Method section that Chapter 48 already reported in detail and into the paper's Introduction, Applications, and Conclusions sections, which that first reading had no occasion to extract in comparable depth. What emerges is not one finding but two, and the second was not anticipated when this chapter was begun. Section 55.2 reports a stated purpose after all, closing the gap Chapter 74 flagged. Section 55.3 reports something Chapter 74 did not ask for: the paper's own Applications section cites, by name, two items of the numerical weather-prediction icosahedral-grid literature (Chapters 51 to 53's subject) as the reason a later design choice was made. That finding requires a further, targeted amendment to Chapter 74's own convergence discussion, made in Section 55.4.

**Methodological Note**

**Source and what changed since Chapter 48.** White (2000) was fetched in full in this pass from the same author-deposited ResearchGate copy Chapter 48 used, cross-checked against the same SpringerLink DOI landing page; this is a second, targeted re-reading of an already-fetched source, not a new document. Heikes and Randall (1995a) is cited here only at the level White (2000)'s own text cites it (author names, year, and the specific finding White attributes to it); Heikes and Randall's paper itself was not independently fetched or read in this pass, and nothing below claims more about it than what White's citation of it states. The same applies to Thuburn (1997) and Sahr and White (1998): both are known to this volume only through White (2000)'s own citation of them, not through independent reading.

## 55.2 White's own stated purpose

White (2000)'s Introduction does state a purpose, in terms no less explicit than the statements this volume already extracted for QTM, HTM, or GBT; Chapter 48's first reading simply had

no occasion to quote the Introduction's framing sentence, having gone directly to the Method section instead. The paper states its own purpose as presenting diamonds as the tessellation unit and demonstrating their properties for grids applied to survey sampling, dynamic modelling, and data storage and retrieval[1]. This is a real purpose statement, but it sits at a different level of generality than the other four: it names the *application domains* diamonds are meant to serve, not a single concrete engineering problem (conservation under integration for Sadourny; measurement-precision bookkeeping for QTM; database query performance for HTM; image-array addressing for GBT) that a specific pre-existing alternative failed to solve.

A second, more specific rationale appears in the paper's Conclusions, and it is closer in kind to the other four chapters' motivations. The paper states that diamond geometry combines properties no single alternative shape combined: it is simpler than triangles or hexagons; its recursive hierarchy is nested, unlike the aperture-3 or aperture-4 hexagon hierarchies Chapter 48 already described; it tessellates the polyhedron completely, unlike the aperture-7 hexagon hierarchy (whose polyhedral tiling White (2000) calls an open conjecture, already reported in Chapter 48's own todobox); and its cells have radial symmetry and translation congruence, properties triangles lack. Read this way, White's originating problem was combinatorial rather than singular: no available tessellation unit combined nesting, complete tessellation, and radial symmetry at once, and the diamond was proposed to fill that specific combinatorial gap. This chapter reports both statements rather than picking one as *the* motivation; the two are complementary (the Introduction states what the method is *for*, the Conclusions state what specific geometric shortfall in existing alternatives it was built *to fix*), not competing readings of the same sentence.

## 55.3 An unlooked-for citation trail into the NWP tradition

The originally scoped task for this chapter ended with Section 55.2. Reading the paper's Applications section in full for that purpose surfaced a second finding this chapter did not set out to look for. White (2000)'s Applications section, discussing why a hexagon tessellation corresponding to the diamond hierarchy is desirable, cites Thuburn (1997) by full title as an example of hexagonal icosahedral grids used in modelling atmospheric fluid dynamics — precisely the numerical-weather-prediction tradition Chapters 51 to 53 examine. The paper then goes further: it reports that Heikes and Randall (1995a) found polar-oriented icosahedron-based hexagon grids produced simulated dynamics asymmetric across the equator, and that better performance followed from a more symmetric grid; it credits Sahr and collaborators (Sahr and White, 1998; and a further, then-in-press Kimerling et al. paper) with proposing the ISEA (Icosahedral Snyder Equal Area) grid specifically to meet that symmetry criterion, by placing edge midpoints rather than vertices at the poles. White (2000) closes this thread by noting that its own diamond subdivision is compatible with the ISEA grid, since ISEA is itself derived from the icosahedron.

This is a materially different kind of connection than the one Chapter 54 reported for the Sadourny/QTM/White cluster as a whole. Chapter 54 showed that a 2003 survey, written by authors including White himself, retrospectively classified the NWP grids and White's diamonds inside one taxonomy — a connection recognized three years *after* White (2000) was published. What this chapter reports is earlier and more direct: White (2000)'s own 2000 text, in constructing its case for why a hexagon tessellation matters and which orientation of that hexagon tessellation to prefer, cites the NWP-adjacent atmospheric modelling literature

[1] White (2000), Introduction. Paraphrased here at slightly more length than the paper's own single sentence; the three application domains named — survey sampling, dynamic modelling, storage/retrieval — match the paper's abstract verbatim.

(Thuburn, 1997; Heikes and Randall, 1995a) and credits Sahr's own NWP-aware symmetry work (Sahr and White, 1998) as the source of a specific design choice. This is White's own paper drawing, at the time of writing, on material connected to the numerical-weather-prediction tradition — not a later survey finding the two traditions taxonomically alike.

One caveat belongs here and is not glossed over. The citation trail runs through Thuburn (1997) and Heikes and Randall (1995a) — both mid-1990s papers building on, but not identical to, the foundational 1968 grids Chapters 51 and 52 examine. White (2000)'s own reference list, read in full in this pass, does not cite Sadourny et al. (1968) or Williamson (1968) directly. What this chapter establishes is that White's diamond paper engages with the *descendant* numerical-weather-prediction literature of the 1990s, not that White read the 1968 founding papers themselves; whether Thuburn (1997) or Heikes and Randall (1995) themselves cite Sadourny or Williamson directly has not been checked in this pass and is left open (Section 55.4).

## 55.4 Consequence: a further, narrower amendment to Chapter 74

Chapter 74, as amended by Chapter 54, already qualifies its independence claim for the Sadourny/QTM/White cluster as independence *of invention*, stating that no evidence in this volume shows White's diamonds' construction drawing on the NWP grids. Section 55.3 shows that statement is now too strong for White specifically. It remains accurate for QTM: no citation trail from Dutton (1999) or the earlier QTM sources examined in Chapter 37 into the NWP literature has been located in this volume. But for White's diamonds, this chapter's reading of the Applications section shows a direct citation link, at the time of original publication, into descendant NWP-tradition literature, used to justify a specific design choice (compatibility with the equatorially symmetric ISEA grid). Chapter 74's own text is amended immediately following this chapter (Chapter 74's Section 74.3) to narrow the invention-independence claim so that it no longer includes White's diamonds without qualification.

**Key Result**

Revisiting White (2000) in full for its own stated purpose, rather than for the construction detail Chapter 48 already extracted, resolves the first open item in Chapter 74's todobox: the paper does state a purpose (survey sampling, dynamic modelling, storage/retrieval) and a more specific combinatorial design rationale (nesting, complete tessellation, and radial symmetry at once, which no single existing alternative shape offered together). The same reading surfaced an unplanned second finding: White's Applications section cites the NWP-descendant atmospheric-modelling literature (Thuburn, 1997; Heikes and Randall, 1995a) and Sahr's own NWP-symmetry-aware work (Sahr and White, 1998) to justify building the diamond scheme compatibly with the equatorially symmetric ISEA grid — a direct, contemporaneous citation link into the NWP tradition, not merely the later shared classification Chapter 54 reported for the cluster as a whole. This narrows, specifically for White's diamonds and not for QTM, the invention-independence claim Chapter 74 makes, in a further amendment made immediately following this chapter.

**To Complete / Verify**

Two items are left open by this chapter. **(i) Whether Thuburn (1997) or Heikes and Randall (1995a) themselves cite Sadourny et al. (1968) or Williamson (1968) directly** has not been checked; this volume knows only that White's 2000 paper cites the 1990s descendant literature, not whether that literature in turn cites the 1968 founding papers by name.

Reading Thuburn (1997) and Heikes and Randall (1995) in full would settle this and is left for a future pass. **(ii) The second item Chapter 74's original todobox left open** — a comparably explicit originating-motivation statement for Vince & Zheng (2009)'s PYXIS algebra, whose full text remains unfetched for the reasons given in Chapter 49 — is untouched by this chapter and remains open exactly as Chapter 74 left it.

Chapter 56

# Closing (Partly) the Citation-Trail Question: Does Thuburn (1997) or Heikes and Randall (1995a,b) Cite Sadourny (1968) or Williamson (1968) Directly?

## 56.1 Scope: the first item left open by Chapter 55's todobox

Chapter 55's closing todobox (Section 55.4) left open, as its first item, whether Thuburn (1997) or Heikes and Randall (1995a) — the two items of NWP-descendant literature that White (2000)'s Applications section itself cites by name — in turn cite Sadourny et al. (1968) or Williamson (1968) directly. This chapter checks that question against each paper's own published reference list, insofar as each is accessible, rather than against any secondary characterization of the two papers' shared lineage. The two cases resolve differently and are reported separately.

## 56.2 Thuburn (1997): resolved

Thuburn (1997)'s own reference list, as indexed by the publisher (American Meteorological Society, *Monthly Weather Review* 125(9)), includes the entry: *"Sadourny, R., A. Arakawa, and Y. Mintz, 1968: Integration of the non-divergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere. Mon. Wea. Rev., 96, 351–356."* This is the same paper catalogued in this volume's bibliography as Sadourny et al. (1968). The entry was located directly on the journal's own article page for Thuburn (1997), not via any third party's paraphrase of what Thuburn cites — the same tier of evidence this volume already applies to a fetched reference list elsewhere (compare Sahr et al., 2003's reference list in Chapter 54), though here obtained through the journal's indexed metadata rather than a full-text PDF fetch, since the AMS platform's full text sits behind the same access barrier already noted at Sadourny et al. (1968)'s own bibliography entry.

**Key Result**

Thuburn (1997) cites Sadourny et al. (1968) directly, by full bibliographic entry, in its own reference list. This resolves half of the first item Chapter 55's todobox left open: the 1990s NWP-descendant literature that White (2000) cites does, in at least this one case, in turn cite one of the two 1968 founding papers by name — not merely share a later commentator's classification of both as the same lineage, which is the weaker claim Chapter 54 had already

established for the cluster as a whole. No equivalent direct citation of Williamson (1968) was found in Thuburn (1997)'s reference list as indexed; its absence is reported here rather than assumed to mean the paper is silent on Williamson's work by some other route (an in-text mention without a bibliography entry, for instance, would not show up in this check).

## 56.3 Heikes and Randall (1995a,b): not resolved

No equivalent direct access to Heikes and Randall (1995a)'s or Heikes and Randall (1995b)'s own reference list was obtained in this pass. The AMS journal platform's article pages for both papers did not surface indexed reference-list content the way Thuburn (1997)'s did, and an author-hosted copy located during this pass sits on a domain that declines automated access (`kiwi.atmos.colostate.edu`), the same kind of barrier already noted for other sources in this bibliography (Sadourny et al., 1968; Sadourny and Morel, 1969).

What this pass did find, repeatedly, across more than a dozen independent later papers spanning 1997 through 2018 (Steppeler et al., 2008[1] and numerous others), is that Heikes and Randall (1995a,b) are routinely listed alongside Sadourny et al. (1968) and Williamson (1968) as the same three-item founding lineage for the icosahedral-hexagonal grid in NWP. This is exactly the weaker, already-established claim: a later commentator's shared classification, not evidence of what Heikes and Randall's own 1995 reference list contains. One paper located in this pass (Walko and Avissar, 2011) goes further, stating that its own grid construction "is similar to methods used in Sadourny et al. (1968), Williamson (1968), Heikes and Randall (1995a)" — a methodological similarity claim by a third party, again not a report of what Heikes and Randall themselves cited.

**To Complete / Verify**

Whether Heikes and Randall (1995a) or Heikes and Randall (1995b) themselves cite Sadourny et al. (1968) or Williamson (1968) directly remains open. This narrows, but does not close, the item Chapter 55's todobox left open: Thuburn (1997)'s side of the question is now settled (Section 56.2); the Heikes-and-Randall side is not, and the accumulated secondary evidence surveyed in Section 56.3, however consistent, is not a substitute for reading the papers' own bibliographies. Obtaining either paper's full text or reference list through a route this volume has not yet tried (a university repository copy, an interlibrary-loan-accessible archive, or a citation database that reproduces full reference lists rather than selected entries) is left for a future pass. The second item Chapter 74's original todobox left open — an originating-motivation statement for Vince & Zheng (2009)'s PYXIS algebra — remains untouched by this chapter and open exactly as Chapter 55 left it.

[1] Not otherwise cited in this volume; noted here only for the shared-lineage grouping it reports.

Chapter 57

# Patents on the Sphere: Cartography as Intellectual Property

## 57.1 Scope: the cost of distortion as a claimed invention

This volume's earlier chapters treat every projection and every DGGS as a mathematical or algorithmic object, judged by what it preserves and what it distorts. This closing historical chapter looks instead at a recurring institutional fact about several of the specific schemes this volume already discusses in detail: at least four of them — Cahill's octahedral Butterfly Map (Chapter 22), Fuller's cuboctahedral Dymaxion map, Sahr's icosahedral MGBT/aperture-3 hexagon tree (Chapter 47), and Peterson's PYXIS multiresolution grid (Chapter 49) — were formally patented by their inventors, in the United States, decades or a century apart. This chapter reads that pattern as evidence for a claim this volume's title puts more abstractly: that reducing the cost of distortion has been treated, repeatedly and across a full century, as a proprietary asset worth the expense of a patent application, not merely as a mathematical curiosity to be published and left free.

**Methodological Note**

This chapter introduces one new source to Section 114.6: Fuller (1946), R. Buckminster Fuller's 1946 U.S. Patent 2,393,676, "Cartography," fetched and read in full via Google Patents. The Cahill, Sahr, and Peterson patent material this chapter draws on is not re-fetched here; it is cited from Chapters 22, 47 and 49 respectively, where the relevant patents (or, for Cahill, the tertiary source reporting the patent) were already fetched, read, and in two cases (Sahr, Peterson) transcribed in structural detail. No new computation is performed; this is a comparative historical synthesis across sources already independently established elsewhere in this volume, plus the one newly fetched primary source.

## 57.2 Fuller's 1946 patent: uniform peripheral scale as the claimed advance

Fuller (1946) was filed on February 25, 1944 and granted on January 29, 1946, expiring, per the patent's own recorded legal status, in 1963. Fuller's own framing, read directly from the patent's text, is worth recording in this volume's terms: he opens by stating that every flat map is necessarily a compromise with the sphere's own truth, naming the Mercator (true to scale only along the Equator) and azimuthal projections (meridian convergence at a single pole) as examples, and criticizes prior gnomonic polyhedral maps for carrying gnomonic projection's own severe radial distortion out to each facet's edges. Fuller's stated object, per Fuller (1946), is a sectional world map whose parts can be reassembled to give a more accurate overall picture of areas, boundaries, directions, and distances than any prior flat map, constructed so that land masses can be joined with fewer of what the patent calls "sinuses" (cuts) than earlier sectional

maps allowed.

**Key Result**

The mechanism Fuller (1946) claims is a specific distortion-placement strategy, not a specific distortion-elimination one, and it is exactly the same strategic move this volume's DGGS chapters make in a different vocabulary. Fuller's polyhedron — six equilateral squares and eight equilateral triangles, all vertices lying on common great circles, which he names a "dymaxion" — is constructed so that every section edge is a straight-line representation of a great circle and therefore carries uniform, undistorted scale; the patent states explicitly that this pushes whatever distortion the construction cannot avoid (Fuller's own term is "subsidence errors") from the periphery of each section toward its interior, rather than requiring any corrective factor. This is, in this volume's own vocabulary from Chapters 30 and 58, a boundary-versus-interior distortion trade rather than a distortion-elimination claim, decades before this volume's own boundary-rate analyses formalize the same trade for DGGS cells.

The patent's three claims, read directly, are narrower than Fuller's own prose description: Claim 1 covers any projected map made of square and triangular sections matching along projected-great-circle edges with uniform-scale delineation along those edges; Claim 2 specifies the two-way (square) and three-way (triangular) great-circle grid construction method; Claim 3 specifies the particular six-square, eight-triangle solid by name. None of the three claims covers the icosahedral twenty-triangle variant Fuller and Shoji Sadao produced in 1954 — the "Dymaxion Airocean World" map most commonly associated with Fuller's name today — a point this chapter flags without resolving, since Fuller (1946) itself, as fetched in this pass, only describes and claims the earlier cuboctahedral (six-square, eight-triangle) form.

**To Complete / Verify**

**Narrowed by convergent secondary evidence, not yet closed by a primary patent-office record.** This chapter had not located a source establishing whether Fuller sought a separate patent covering the 1954 icosahedral Airocean World construction. Several independent rare-map-dealer provenance accounts, each researched separately for sale listings of the physical 1954 map (Boston Rare Maps; Ruderman Maps, two listings), converge on the same patent history without prompting: a single application, filed February 1944 and granted January 1946, covering the earlier *cuboctahedral* Dymaxion version shown in Life (1943)'s photographic essay; none of these accounts mentions any second filing when narrating the 1954 icosahedral map's own publication, describing it consistently as a redesign ("revamped," "dropping the cuboctahedron and adopting a modified icosahedron") published under the existing Dymaxion trademark rather than as a newly patented construction. This is suggestive convergence across independently written sources, not a USPTO record itself, so it narrows rather than closes the question; a direct search of U.S. patent records for any Fuller or Fuller-and-Sadao filing dated 1954 or later remains the more authoritative step and has still not been carried out here.

## 57.3 A century of projection and DGGS patents, compared

Reading Fuller (1946) against the three other patented schemes this volume already documents in detail gives a rough timeline this chapter did not set out to construct but that falls out of the sources already gathered. Cahill's octahedral projection was patented in 1913, per the tertiary source Chapter 22 already relies on for that figure; Fuller's cuboctahedral map, in

1944–1946; and, sixty years later, two DGGS schemes this volume reproduces computationally in detail were themselves filed as patents rather than (or in addition to) published as papers: Peterson's PYXIS grid, with a filing date of 2004 and a stated priority date of 2003, granted in 2011 (Peterson, 2011); and Sahr's icosahedral Modified Generalized Balanced Ternary and Aperture 3 Hexagon Tree, filed in January 2005 and granted in January 2011 (Sahr, 2005/2011). Chapters 47 and 49 both note, independently of this chapter, that the patent text in each of these two more recent cases turned out to be the more complete and more directly usable technical source available for this volume's own reproduction work, in each case more so than the corresponding peer-reviewed journal article by the same author or authors.

**Key Result**

Read together, these four patents span from 1913 to 2011 and cover every polyhedral family this volume treats as historically important: octahedral (Cahill), a mixed cuboctahedral square-and-triangle solid (Fuller), and two distinct icosahedral hierarchical schemes (Sahr, Peterson). That every one of these four inventors chose to patent rather than (or in addition to) publish is this chapter's central empirical observation, and it bears directly on this volume's own title: distortion-reduction schemes for the sphere have been treated, across a full century and independently by at least four separate inventors working in unrelated decades and for unrelated purposes (a geography-magazine contributor, an architect-inventor, a government geographer, and a commercial digital-earth company), as inventions with commercial or institutional value worth the specific expense and public disclosure of a patent filing — not merely as mathematical results whose value lies only in being correct and published.

## 57.4 What a projection patent can and cannot claim

Reading Fuller (1946)'s own claims language against Peterson (2011)'s and Sahr (2005/2011)'s (as those two are already characterized in Chapters 47 and 49) surfaces a structural constraint common to all three: a patent on a map projection or a DGGS cannot claim the underlying geometric or number-theoretic fact the scheme depends on (that an octahedron, cuboctahedron, or icosahedron can be inscribed in a sphere with all vertices on great circles; that a hexagonal aperture-3 or aperture-4 hierarchy has certain addition-table or self-similarity properties), since such facts are mathematical and not patentable subject matter in the relevant sense; each patent instead claims a specific *construction method* or a specific *combination of named solid, grid method, and stated scale-uniformity property*, exactly the scope Fuller (1946)'s own three claims stay within. This is consistent with, though this chapter does not treat it as proof of, the pattern Chapter 49 already reports: that the patent language covering PYXIS and MGBT reads as an engineering specification (addition tables, worked numeric examples, explicit algorithmic steps) rather than as a mathematical proof, the same register Fuller (1946)'s own description of the great-circle grid construction device (its numbered points, hemisphere member, and step-by-step tracing procedure) is written in.

## 57.5 Closing: an intellectual-property genealogy alongside the mathematical one

Chapter 35 traces this volume's mathematical genealogy of the cost of distortion from Tissot through Peters to Goldberg-Gott; Chapters 2, 3 and 7 trace three further historical precedents

for that same cost, in discrete addressing, regional zoning, and explicit multi-property compromise, respectively. This chapter's genealogy runs alongside all four, in a different register: not who proved what about distortion, but who filed to own a specific engineered response to it, and when. That the same underlying problem — representing a sphere's surface with controlled, bounded distortion — has drawn patent filings in 1913, 1944, 2003/2004, and 2005, from inventors as different as a geography-magazine contributor, an architect, and two separate twenty-first-century digital-earth engineers, is this volume's own best evidence that the cost of distortion has never been treated, by the people closest to the problem, as merely an academic curiosity.

Chapter 58

# Extending EVT Boundary Laws to the Discrete Case

## 58.1 What kind of boundary law is needed here

*No Map Is Best*'s latitude-band chapters (Ch. 24 and Ch. 30–34) characterized how a continuous distortion functional $\delta$ behaves as a sampling window approaches a boundary of its domain: a survival rate $S(D-\varepsilon) \sim K\varepsilon^p$ as $\varepsilon \to 0^+$, with the exponent $p$ diagnostic of the boundary's geometry (linear for an edge, quadratic for a corner). The discrete graticule of Chapter 1 has its own boundary: the pole, $\phi \to \pi/2$. This chapter asks the same question of that boundary — at what rate does the ring-area ratio $\rho_i$ of Proposition 1.2 degenerate as the resolution parameter $K$ grows, and does this rate differ from the *typical* (bulk) discretization error found away from the pole?

Two results are proved below. The first (Proposition 58.1) is the discrete boundary law itself, exact rather than asymptotic: the worst-case (polar) cell-area ratio has closed form $\rho_{\min}(K) = \tan(\pi/4K)$, decaying as $K^{-1}$. The second (Proposition 58.3) bounds the *uniform* error between the discrete law and its continuous $\cos\phi$ limit, and shows this error decays as $K^{-2}$ — one order faster than the boundary rate. The gap between these two rates is itself the discrete analogue of what *No Map Is Best* called a genuine boundary singularity: the pole is not just where $\rho_i$ happens to be smallest, but where the discretization error decays at a structurally slower rate than everywhere else on the graticule.

## 58.2 The exact boundary law

**Proposition 58.1** (Exact minimum ring-area ratio)**.** *Under Definition 1.1, $\rho_i$ of Eq. (1.2) is strictly decreasing in $i$ on $\{0,\dots,K-1\}$, and its minimum, attained at the pole-adjacent ring $i = K-1$, is*

$$\rho_{\min}(K) \;=\; \rho_{K-1} \;=\; \tan\!\left(\frac{\Delta\phi}{2}\right) \;=\; \tan\!\left(\frac{\pi}{4K}\right). \tag{58.1}$$

*Proof.* Monotonicity: from Eq. (1.2), $\rho_i \propto \cos\big((i+\tfrac{1}{2})\Delta\phi\big)$, and cos is strictly decreasing on $\big[0, \pi/2 - \Delta\phi/2\big]$, the range spanned by $(i+\tfrac{1}{2})\Delta\phi$ for $i = 0,\dots,K-1$; hence $\rho_i$ is strictly decreasing in $i$, and its minimum over $i \in \{0,\dots,K-1\}$ is attained at $i = K-1$.

For the closed form, set $i = K-1$ in Eq. (1.2): $(i+\tfrac{1}{2})\Delta\phi = (K-\tfrac{1}{2})\Delta\phi = \tfrac{\pi}{2} - \tfrac{\Delta\phi}{2}$ (using $K\Delta\phi = \pi/2$). Then $\cos\big(\tfrac{\pi}{2} - \tfrac{\Delta\phi}{2}\big) = \sin(\Delta\phi/2)$, so

$$\rho_{K-1} = \frac{\sin(\Delta\phi/2)}{\cos(\Delta\phi/2)} = \tan(\Delta\phi/2),$$

which with $\Delta\phi = \pi/(2K)$ gives Eq. (58.1). □

*Remark* 58.2 (Reading the boundary law). Equation (58.1) is exact at every finite $K$, not merely an asymptotic statement; its leading behaviour as $K \to \infty$ is $\rho_{\min}(K) = \frac{\pi}{4K} + O(K^{-3})$, since $\tan x = x + x^3/3 + O(x^5)$ for small $x$. The exponent is 1 (a linear rate in $1/K$), matching the *edge*-type boundary rate of *No Map Is Best*, Ch. 24, rather than the quadratic *corner* rate of Ch. 31: the polar ring is reached along a single one-dimensional family of rings (a genuine edge of the $[0, \pi/2]$ colatitude interval), not by two coordinates closing simultaneously, so a linear rate is the structurally expected outcome, confirmed here in exact rather than asymptotic form.

## 58.3 The uniform (bulk) rate, and why it differs

**Proposition 58.3** (Uniform discretization error). *Under Definition 1.1, writing $\phi_i := (i + \frac{1}{2})\Delta\phi$ for the midpoint colatitude of ring $i$,*

$$\rho_i - \cos\phi_i \;=\; \cos\phi_i \,[\sec(\Delta\phi/2) - 1]\,, \qquad i = 0, \dots, K-1, \tag{58.2}$$

*and hence*

$$\sup_i \big|\rho_i - \cos\phi_i\big| \;\le\; \sec(\Delta\phi/2) - 1 \;=\; \frac{(\Delta\phi)^2}{8} + O\big((\Delta\phi)^4\big). \tag{58.3}$$

*Proof.* From Eq. (1.2), $\rho_i = \cos\phi_i \sec(\Delta\phi/2)$, so $\rho_i - \cos\phi_i = \cos\phi_i\big[\sec(\Delta\phi/2) - 1\big]$, which is Eq. (58.2). Since $\sec(\Delta\phi/2) - 1 \ge 0$ for $\Delta\phi \in [0, \pi)$ and $|\cos\phi_i| \le 1$ for every $i$, taking the supremum over $i$ gives the bound in Eq. (58.3), and the bound is attained in the limit $\phi_i \to 0$ (equatorial ring). The Taylor expansion $\sec x = 1 + x^2/2 + O(x^4)$ at $x = \Delta\phi/2$ gives $\sec(\Delta\phi/2) - 1 = (\Delta\phi)^2/8 + O\big((\Delta\phi)^4\big)$. □

**Key Result**

Two distinct, exact convergence rates coexist in the same discrete graticule: the *typical* (uniform, bulk) discretization error against the continuous $\cos\phi$ law decays as $\Theta(K^{-2})$ (Proposition 58.3), while the *extreme* (polar, boundary) cell-area ratio itself decays only as $\Theta(K^{-1})$ (Proposition 58.1). The pole is therefore a genuine boundary singularity in exactly the EVT sense used throughout *No Map Is Best*: refining the grid improves the typical cell one order of magnitude faster than it improves the worst cell. This is the discrete counterpart of the empirical regularity (regional optimization beating global compromise) that motivates Part II below.

## 58.4 Numerical verification

Table 58.1 confirms both rates directly: column "ratio (a)" is $\rho_{\min}(K)$ divided by its linear approximation $\pi/(4K)$, and should tend to 1; column "ratio (b)" is the numerically computed $\sup_i |\rho_i - \cos\phi_i|$ divided by the quadratic approximation $(\Delta\phi)^2/8$, and should likewise tend to 1, at a visibly faster rate (since the next-order correction terms are $O(K^{-2})$ and $O(K^{-4})$ respectively, matching the two propositions' proofs).

Both ratio columns converge to 1 from above (ratio (a)) and below (ratio (b)), as the sign of the respective next-order correction terms in the Taylor expansions predicts; neither is fitted post-hoc, since both approximations were fixed analytically *before* the numerical check was run.

Table 58.1: Two convergence rates in the $K$-ring discrete graticule. Column (a): exact minimum ratio $\rho_{\min}(K) = \tan(\pi/4K)$ against its linear approximation $\pi/(4K)$. Column (b): numerically computed $\sup_i |\rho_i - \cos\phi_i|$ against its quadratic approximation $(\Delta\phi)^2/8$. Source: `code/evt_discrete_boundary_check.py`, `data/evt_discrete_boundary_results.json`.

| | (a) Boundary rate | | | (b) Bulk rate | | |
|---|---|---|---|---|---|---|
| $K$ | $\rho_{\min}(K)$ | $\pi/(4K)$ | ratio | $\sup_i \lvert\rho_i - \cos\phi_i\rvert$ | $(\Delta\phi)^2/8$ | ratio |
| 4 | $1.9891 \times 10^{-1}$ | $1.9635 \times 10^{-1}$ | 1.01305 | $1.9215 \times 10^{-2}$ | $1.9277 \times 10^{-2}$ | 0.99679 |
| 8 | $9.8491 \times 10^{-2}$ | $9.8175 \times 10^{-2}$ | 1.00323 | $4.8153 \times 10^{-3}$ | $4.8191 \times 10^{-3}$ | 0.99920 |
| 16 | $4.9127 \times 10^{-2}$ | $4.9087 \times 10^{-2}$ | 1.00080 | $1.2045 \times 10^{-3}$ | $1.2048 \times 10^{-3}$ | 0.99980 |
| 32 | $2.4549 \times 10^{-2}$ | $2.4544 \times 10^{-2}$ | 1.00020 | $3.0118 \times 10^{-4}$ | $3.0120 \times 10^{-4}$ | 0.99995 |
| 64 | $1.2272 \times 10^{-2}$ | $1.2272 \times 10^{-2}$ | 1.00005 | $7.5298 \times 10^{-5}$ | $7.5299 \times 10^{-5}$ | 0.99999 |
| 128 | $6.1360 \times 10^{-3}$ | $6.1359 \times 10^{-3}$ | 1.00001 | $1.8825 \times 10^{-5}$ | $1.8825 \times 10^{-5}$ | 1.00000 |
| 256 | $3.0680 \times 10^{-3}$ | $3.0680 \times 10^{-3}$ | 1.00000 | $4.7062 \times 10^{-6}$ | $4.7062 \times 10^{-6}$ | 1.00000 |
| 512 | $1.5340 \times 10^{-3}$ | $1.5340 \times 10^{-3}$ | 1.00000 | $1.1765 \times 10^{-6}$ | $1.1765 \times 10^{-6}$ | 1.00000 |
| 1024 | $7.6699 \times 10^{-4}$ | $7.6699 \times 10^{-4}$ | 1.00000 | $2.9414 \times 10^{-7}$ | $2.9414 \times 10^{-7}$ | 1.00000 |

## 58.5 Scope and open extensions

**To Complete / Verify**

This chapter treats only the equiangular graticule of Definition 1.1. Two extensions were originally left open here: (i) a general (non-equiangular) ring placement $0 = \phi_0 < \cdots < \phi_K = \pi/2$, for which Eq. (1.1) still holds ring-by-ring but Eq. (1.2)'s closed form does not, and the boundary rate would depend on how the ring boundaries are allowed to cluster near the pole; (ii) an actual DGGS (H3, S2, rHEALPix, DGGRID, OpenEAGGR) rather than a toy lat/lon graticule, for which the analogous boundary statistic would need to be computed numerically from the cell geometries themselves, along the lines of Kmoch et al. (2022), rather than derived in closed form. Item (i) is now addressed, for an explicit one-parameter family of placements, in Chapter 60, which finds a sharp phase transition in the boundary rate rather than a mere rescaling; item (ii) is now addressed, for H3 and S2, in Chapter 61, once Chapters 11 and 13 supplied the needed empirical figures — with the qualified conclusion that real DGGS do not obey a boundary-decay law of this kind at all, for reasons sketched but not proved there.

Chapter 59

# The Dual Scheme, and an Impossibility Theorem

## 59.1 Why a dual scheme

Chapter 58 derived an exact boundary law for the *equiangular* graticule of Definition 1.1: equal angular width forces cell area to collapse toward the pole, at rate $\Theta(K^{-1})$. That result invites an obvious question, not asked in *No Map Is Best* because it has no continuous analogue: what happens under the opposite convention, in which ring boundaries are chosen so that every ring has *exactly* equal area instead of equal angular width? This chapter answers that question with its own exact boundary law (Section 59.2), and then proves a general theorem (Section 59.3) showing that the two conventions' respective degeneracies are not a coincidence of these two particular constructions: no ring-partition scheme, of any kind, can avoid both.

## 59.2 The equal-area scheme and its own boundary law

**Definition 59.1** (Equal-area $K$-ring graticule)**.** Divide the quadrant $\phi \in [0, \pi/2]$ into $K$ rings by setting $\phi_i := \arcsin(i/K)$ for $i = 0, \ldots, K$, so that ring $i$ is $[\phi_i, \phi_{i+1})$. As in Definition 1.1, use a constant longitude step $\Delta\lambda$ within each ring.

**Proposition 59.2** (Exact equal-area property)**.** *Under Definition 59.1, every ring has exactly the same physical area: $A_i = R^2\Delta\lambda/K$ for all $i = 0, \ldots, K-1$, independent of $i$.*

*Proof.* By the area formula already used in the proof of Proposition 1.2, $A_i = R^2\Delta\lambda(\sin\phi_{i+1} - \sin\phi_i)$. Under Definition 59.1, $\sin\phi_i = i/K$ by construction, so $\sin\phi_{i+1} - \sin\phi_i = (i+1)/K - i/K = 1/K$ for every $i$, giving $A_i = R^2\Delta\lambda/K$ regardless of $i$. □

This is the exact discrete analogue of a sinusoidal-type equal-area graticule; Proposition 59.2 is not new to cartography, but stating it as an exact, finite-$K$ proposition — rather than invoking it only in the continuum — is what allows its own boundary law to be derived with the same rigor as Chapter 58's equiangular one.

**Proposition 59.3** (Angular boundary law for the equal-area scheme)**.** *Under Definition 59.1, the angular width $\Delta\phi_i := \phi_{i+1} - \phi_i$ is strictly increasing in $i$, and the polar ring's width satisfies*

$$\Delta\phi_{K-1} \;=\; \frac{\pi}{2} - \arcsin\!\left(1 - \frac{1}{K}\right) \;=\; \sqrt{\frac{2}{K}} \;+\; O\!\left(K^{-1}\right) \qquad \text{as } K \to \infty. \tag{59.1}$$

*Proof.* Write $\Delta\phi_i = \arcsin\!\left(\frac{i+1}{K}\right) - \arcsin\!\left(\frac{i}{K}\right)$. Since arcsin is strictly convex on $[0, 1)$ (its second derivative $x(1-x^2)^{-3/2}$ is positive for $x \in (0,1)$), equal increments in its argument produce strictly increasing increments in its value, so $\Delta\phi_i$ is strictly increasing in $i$.

For the polar ring, $i = K-1$: $\Delta\phi_{K-1} = \arcsin(1) - \arcsin(1-1/K) = \frac{\pi}{2} - \arcsin(1-1/K)$, which is exact and requires no further argument for the first equality of Eq. (59.1). For the asymptotic rate, set $\arcsin(1-\varepsilon) = \frac{\pi}{2} - \delta$ with $\varepsilon = 1/K \to 0$; then $\cos\delta = \sin\big(\arcsin(1-\varepsilon)\big) = 1-\varepsilon$, and expanding $\cos\delta = 1 - \delta^2/2 + O(\delta^4)$ gives $\delta^2/2 = \varepsilon + O(\delta^4)$, so $\delta = \sqrt{2\varepsilon} + O(\varepsilon^{3/2})$. With $\varepsilon = 1/K$, $\delta = \sqrt{2/K} + O(K^{-3/2})$, which is $\Delta\phi_{K-1} = \sqrt{2/K} + O(K^{-1})$ after matching orders (the $O(\varepsilon^{3/2}) = O(K^{-3/2})$ term is dominated by the $O(K^{-1})$ term already present from the next order of the $\delta^4$ correction, so the bound is stated conservatively as $O(K^{-1})$). □

*Remark* 59.4 (Reading the dual boundary law). The mean angular width across all $K$ rings is $\pi/(2K) = \Theta(K^{-1})$. Equation (59.1) shows the polar ring's width is $\Theta(K^{-1/2})$ — asymptotically *larger* than the mean by a factor growing as $\Theta(K^{1/2})$. This is the equal-area scheme's own boundary singularity, structurally dual to Chapter 58's: there, equal angle forced area to vanish at rate $K^{-1}$ relative to a constant mean; here, equal area forces angular width to *diverge in relative terms* at rate $K^{1/2}$ relative to its own mean. Both are exact, both are boundary-localized at the pole, and, as Theorem 59.6 below shows, both are instances of the same underlying obstruction.

Table 59.1 verifies Eq. (59.1) numerically: the ratio of the exact polar width to the asymptotic prediction $\sqrt{2/K}$ tends to 1, while the ratio of the same quantity to the *mean* width $\pi/(2K)$ diverges, growing essentially as $\sqrt{K}$.

Table 59.1: The equal-area scheme's own boundary law. Column "ratio to mean" is $\Delta\phi_{K-1}$ divided by the mean width $\pi/(2K)$ (diverges); column "ratio to asym. pred." is $\Delta\phi_{K-1}$ divided by $\sqrt{2/K}$ (converges to 1). Source: `code/dual_scheme_impossibility_check.py`, `data/dual_scheme_impossibility_results.json`.

| $K$ | $\Delta\phi_{K-1}$ | mean $\Delta\phi$ | ratio to mean | ratio to asym. pred. |
|---|---|---|---|---|
| 4 | 0.72273 | 0.39270 | 1.840 | 1.02210 |
| 8 | 0.50536 | 0.19635 | 2.574 | 1.01072 |
| 16 | 0.35542 | 0.09817 | 3.620 | 1.00528 |
| 32 | 0.25066 | 0.04909 | 5.106 | 1.00262 |
| 64 | 0.17701 | 0.02454 | 7.212 | 1.00131 |
| 128 | 0.12508 | 0.01227 | 10.193 | 1.00065 |
| 256 | 0.08842 | 0.00614 | 14.410 | 1.00033 |
| 512 | 0.06251 | 0.00307 | 20.375 | 1.00016 |
| 1024 | 0.04420 | 0.00153 | 28.813 | 1.00008 |

## 59.3 An impossibility theorem

Chapter 58 and Section 59.2 exhibit two specific schemes, each with its own boundary degeneracy. The next result shows this is forced: no scheme in the whole family of ring partitions can be free of both.

**Definition 59.5** (General ring-partition scheme)**.** A ring-partition scheme of order $K$ is any strictly increasing sequence $0 = \phi_0 < \phi_1 < \cdots < \phi_K = \pi/2$. Its *area sequence* is $a_i := \sin\phi_{i+1} - \sin\phi_i$ and its *angle sequence* is $\ell_i := \phi_{i+1} - \phi_i$, for $i = 0, \ldots, K-1$.

**Theorem 59.6** (Equiangularity and equal area are incompatible for $K \geq 2$)**.** *For $K \geq 2$, there is no ring-partition scheme (Definition 59.5) whose area sequence $(a_i)$ and angle sequence $(\ell_i)$ are* both *constant in $i$.*

*Proof.* Suppose, for contradiction, that a scheme has $\ell_i \equiv \pi/(2K)$ (constant angle) for all $i$; this forces $\phi_i = i\pi/(2K)$, the equiangular scheme of Definition 1.1, uniquely (a strictly increasing sequence with constant increment summing to $\pi/2$ over $K$ steps has no other solution). Its area sequence is then $a_i = \sin\big((i+1)\pi/(2K)\big) - \sin\big(i\pi/(2K)\big)$, which by Proposition 1.2 satisfies $a_i \propto \rho_i$ and is strictly decreasing in $i$ by Proposition 58.1's proof; in particular $a_0 \neq a_{K-1}$ for $K \geq 2$ (explicitly, $a_0/a_{K-1} = 1/\rho_{K-1} = \cot(\pi/4K) > 1$ for every finite $K$), so $(a_i)$ is not constant. Hence no scheme has both sequences constant, since assuming the angle sequence constant already forces the area sequence to be non-constant.

Symmetrically (and redundantly, but worth recording for the geometric picture it gives), suppose instead $a_i \equiv 1/K$ (constant area); by Proposition 59.2 this forces $\phi_i = \arcsin(i/K)$ uniquely (a strictly increasing sequence with $\sin\phi_i$ having constant increment $1/K$ has no other solution, since arcsin is a bijection on $[0,1]$). Its angle sequence is then $\ell_i = \arcsin\big(\frac{i+1}{K}\big) - \arcsin\big(\frac{i}{K}\big)$, strictly increasing in $i$ by the convexity argument in the proof of Proposition 59.3; in particular $\ell_0 \neq \ell_{K-1}$ for $K \geq 2$, so $(\ell_i)$ is not constant.

Either route rules out the joint hypothesis, proving the theorem. □

*Remark* 59.7 (Where the obstruction actually lives). The proof identifies the obstruction precisely: it is the strict concavity of sin (equivalently, the strict convexity of arcsin) on the open interval $(0, \pi/2)$. A constant-increment input to a strictly concave, strictly increasing function cannot produce a constant-increment output, and vice versa via the inverse function's strict convexity. This is the same elementary mechanism, applied twice from opposite directions, that produces both Chapter 58's boundary law and Section 59.2's dual boundary law: they are not two independent empirical observations about two particular schemes, but two faces of a single inequality.

**Key Result**

Equal angular width and equal cell area are provably incompatible for any ring-partition scheme with $K \geq 2$ rings (Theorem 59.6), and the two specific schemes examined in this volume realize this incompatibility at two different, exactly quantified rates: $\Theta(K^{-1})$ area collapse under equiangularity (Chapter 58), and $\Theta(K^{1/2})$ relative angular blow-up under equal area (Section 59.2). This impossibility theorem is the non-toy foundation on which Chapter 92's Pareto illustration rests: the trade-off examined there is not merely plausible or illustrated by one stylized cost pair, but a consequence of an elementary, general, and now proved obstruction.

## 59.4 A quantitative floor: the universal density-spread theorem

Theorem 59.6 is an existence/nonexistence statement: it rules out perfect simultaneous constancy of $(a_i)$ and $(\ell_i)$, but says nothing about *how far* from constant a scheme is forced to be. This section answers that question — open item (i) of the todobox that originally closed this chapter — for a specific, natural measure of non-constancy, and shows the resulting bound is not merely a property of the two schemes already studied but a floor holding uniformly over *every* scheme of *every* order $K \geq 2$.

**Definition 59.8** (Local density)**.** For a ring-partition scheme (Definition 59.5), the *local density* of ring $i$ is $\delta_i := a_i/\ell_i$.

*Remark* 59.9 ($\delta_i$ is an exact mean value, not an approximation). Since $a_i = \sin\phi_{i+1} - \sin\phi_i = \int_{\phi_i}^{\phi_{i+1}} \cos\phi \, d\phi$, the quantity $\delta_i = a_i/\ell_i$ is, by definition, the mean value of $\cos\phi$ over ring $i$ —

exactly, at every finite $K$, by the integral mean value theorem, not a first-order or small-angle approximation. In particular $\delta_i \in (0,1)$ always, and $\delta_i$ recovers $\cos\phi$ pointwise in the $K \to \infty$ limit, consistently with Remark 1.3.

**Definition 59.10** (Density spread)**.** For a scheme of order $K$, the *density spread* is $D := \delta_0 - \delta_{K-1}$: the gap between the equatorial ring's mean density and the polar ring's.

By Theorem 59.6, $D > 0$ for every scheme (equal density in every ring would itself already force $a_i$ and $\ell_i$ non-constant in a coupled way; the case $D = 0$ specifically is excluded a fortiori by the argument below, which gives a much stronger, strictly positive, $K$-independent bound).

### 59.4.1 The two-ring case

For $K = 2$, a scheme is a single interior boundary $\phi_1 = x \in (0, \pi/2)$, with ring $0 = [0, x]$ and ring $1 = [x, \pi/2]$. By Definition 59.8,

$$\delta_0(x) = \frac{\sin x}{x}, \qquad \delta_1(x) = \frac{1 - \sin x}{\pi/2 - x}, \qquad \text{spread}(x) := \delta_0(x) - \delta_1(x). \tag{59.2}$$

**Proposition 59.11** (Exact boundary values of spread$(x)$)**.** $\lim_{x \to 0^+} \text{spread}(x) = 1 - \frac{2}{\pi}$ *and* $\lim_{x \to (\pi/2)^-} \text{spread}(x) = \frac{2}{\pi}$.

*Proof.* As $x \to 0^+$, $\delta_0(x) = \sin x / x \to 1$ (standard limit), and $\delta_1(x) \to (1-0)/(\pi/2 - 0) = 2/\pi$ directly by substitution (no indeterminate form), giving spread$(x) \to 1 - 2/\pi$. As $x \to (\pi/2)^-$, $\delta_0(x) \to \sin(\pi/2)/(\pi/2) = 2/\pi$ directly by substitution, while $\delta_1(x)$ is $0/0$; write $u = \pi/2 - x \to 0^+$, so $\delta_1 = (1 - \cos u)/u \to 0$ (again standard), giving spread$(x) \to 2/\pi - 0 = 2/\pi$. □

Proposition 59.11 identifies $1 - 2/\pi \approx 0.363380$ as the candidate floor: the two boundary values of spread$(x)$ are $1 - 2/\pi$ and $2/\pi$, and $1 - 2/\pi < 2/\pi$. What remains is to show (a) spread$(x)$ does not dip below $1 - 2/\pi$ anywhere in the interior of $(0, \pi/2)$, and (b) the same floor holds for every $K \geq 2$, not just $K = 2$. Part (b) is established first, in closed form, because it reduces (a) at general $K$ to (a) at $K = 2$ exactly.

### 59.4.2 Reduction of every $K$ to the two-ring case

**Lemma 59.12** (Monotonicity of the two boundary densities)**.** *On $(0, \pi/2)$: (i) $\delta_0(t) := \sin t / t$ is strictly decreasing in $t$; (ii) $\delta_{\text{last}}(t) := (1 - \sin t)/(\pi/2 - t)$ is strictly decreasing in $t$.*

*Proof.* (i) $\delta_0'(t) = (t\cos t - \sin t)/t^2$. Since $\tan t > t$ for $t \in (0, \pi/2)$ (a standard inequality: $g(t) := \tan t - t$ has $g(0) = 0$ and $g'(t) = \sec^2 t - 1 = \tan^2 t > 0$ on $(0, \pi/2)$), $t\cos t - \sin t = \cos t\,(t - \tan t) < 0$, so $\delta_0'(t) < 0$.

(ii) Substitute $u := \pi/2 - t \in (0, \pi/2)$, so $\sin t = \cos u$ and $\delta_{\text{last}} = (1 - \cos u)/u =: s(u)$, with $u$ strictly decreasing in $t$. It suffices to show $s(u)$ is strictly *increasing* in $u$ on $(0, \pi/2)$ (so that $\delta_{\text{last}}(t) = s(\pi/2 - t)$ is strictly decreasing in $t$). Now $s'(u) = [u \sin u - (1 - \cos u)]/u^2$; let $h(u) := u \sin u - (1 - \cos u)$, so $h(0) = 0$ and $h'(u) = \sin u + u\cos u - \sin u = u \cos u > 0$ on $(0, \pi/2)$, hence $h(u) > 0$ there, hence $s'(u) > 0$. □

**Theorem 59.13** (Universal density-spread floor)**.** *For every $K \geq 2$ and every ring-partition scheme of order $K$ (Definition 59.5), the density spread satisfies*

$$D = \delta_0 - \delta_{K-1} \;>\; 1 - \frac{2}{\pi}, \tag{59.3}$$

*and $1 - 2/\pi$ is the exact infimum of $D$ over all $K \geq 2$ and all schemes: no larger constant makes Eq. (59.3) true for every scheme of every order.*

*Proof.* $D$ depends on the scheme only through the two boundary points $\phi_1$ (which fixes $\delta_0$) and $\phi_{K-1}$ (which fixes $\delta_{K-1}$), via $\delta_0 = \delta_0(\phi_1)$ and $\delta_{K-1} = \delta_{\text{last}}(\phi_{K-1})$ in the notation of Lemma 59.12, since $a_0 = \sin\phi_1 - \sin 0 = \sin\phi_1$, $\ell_0 = \phi_1$, and $a_{K-1} = 1 - \sin\phi_{K-1}$, $\ell_{K-1} = \pi/2 - \phi_{K-1}$, irrespective of $K$ and of how the intermediate boundaries $\phi_2, \ldots, \phi_{K-2}$ are placed. By Definition 59.5, $0 < \phi_1 \leq \phi_{K-1} < \pi/2$ for every $K \geq 2$ (equality only possible when $K = 2$, where $\phi_1 = \phi_{K-1}$ by definition; for $K \geq 3$ the inequality is strict but $\phi_1$ and $\phi_{K-1}$ can be brought arbitrarily close by shrinking the intermediate rings).

By Lemma 59.12, $\delta_0(\phi_1)$ is minimized (for fixed upper bound on $\phi_1$) by taking $\phi_1$ as large as the constraint allows, and $\delta_{\text{last}}(\phi_{K-1})$ is maximized (for fixed lower bound on $\phi_{K-1}$) by taking $\phi_{K-1}$ as small as the constraint allows. Since both extremes are achieved jointly at $\phi_1 = \phi_{K-1} =: t$, and the constraint $\phi_1 \leq \phi_{K-1}$ permits this for every $t \in (0, \pi/2)$ (exactly at $K = 2$, and in the limit of shrinking intermediate rings for $K \geq 3$),

$$\inf_{K \geq 2} \; \inf_{\text{schemes of order } K} D \;=\; \inf_{t \in (0,\pi/2)} \big[\delta_0(t) - \delta_{\text{last}}(t)\big] \;=\; \inf_{t \in (0,\pi/2)} \text{spread}(t), \tag{59.4}$$

i.e. the infimum over every scheme of every order equals the infimum of the $K = 2$ function Eq. (59.2) already introduced in Section 59.4.1. Proposition 59.11 gives $\text{spread}(t) \to 1 - 2/\pi$ as $t \to 0^+$, so the right-hand side of Eq. (59.4) is at most $1 - 2/\pi$; Section 59.4.3 establishes that it is not less, so it equals $1 - 2/\pi$ exactly, and since the value $1 - 2/\pi$ is a limit as $t \to 0^+$ and not attained at any interior $t$ (nor, for $K \geq 3$, exactly reachable, since $\phi_1 = \phi_{K-1}$ requires the intermediate rings to have zero width), the inequality in Eq. (59.3) is strict for every actual scheme. □

*Remark* 59.14 (What the reduction buys). Theorem 59.13 converts an infinite-dimensional question — over all $K$ and all interior placements $\phi_2, \ldots, \phi_{K-2}$ — into a one-variable question about a single function on $(0, \pi/2)$, and it does so *exactly*, with no approximation: the reduction Eq. (59.4) is an equality, not a bound. The only remaining work is establishing the infimum of that one-variable function, done next.

### 59.4.3 Establishing the floor of the two-ring function

Proposition 59.11 pins down the two boundary values of $\text{spread}(x)$ exactly. What is not yet shown in closed form is that $\text{spread}(x)$ does not fall below $1 - 2/\pi$ somewhere in the open interval $(0, \pi/2)$. The derivative $\text{spread}'(x)$ has an exact but algebraically unwieldy closed form (a quotient with numerator $-2\pi x^2 \cos x - 4x^2 + 4\pi x \sin x + \pi^2 x \cos x - \pi^2 \sin x$ over $x^2(2x - \pi)^2$), from which a sign argument valid on the whole interval is not immediate. Table 59.2 reports a dense numerical certificate in place of that closed-form step.

The minimum margin in Table 59.2 is attained at an $x$ within one part in $10^6$ of the left endpoint and shrinks toward $0$ exactly as Proposition 59.11 predicts — there is no interior dip; the same run finds zero adjacent-step violations of monotonicity across the full $2 \times 10^6$-point grid, consistent with (though not a proof of) the strictly-increasing behaviour already visible in the coarser 11-point scan of Table 59.3, and consistent with the multi-$K$ check of Table 59.4,

Table 59.2: Dense numerical certificate for spread$(x) > 1 - 2/\pi$ and for strict monotonicity, evaluated at 2,000,000 equally spaced points strictly inside $(0, \pi/2)$ (endpoints excluded). Source: `code/density_spread_bound.py`, `data/density_spread_bound_results.json`.

| Quantity | Value |
|---|---|
| Points evaluated | 2,000,000 |
| Violations of spread$(x) > 1 - 2/\pi$ | 0 |
| Minimum margin over the floor | $1.82 \times 10^{-7}$ |
| $x$ at minimum margin | $7.85 \times 10^{-7}$ |
| Non-monotone adjacent steps | 0 |

which applies the same collapsing-boundary construction used in the proof of Theorem 59.13 directly at $K = 3, 5, 10$ and confirms the margin over the floor shrinks the same way as the collapse point approaches 0, regardless of $K$.

Table 59.3: spread$(x)$ at representative points of $(0, \pi/2)$, confirming the values move monotonically from just above $1 - 2/\pi$ to just below $2/\pi$. Source: `code/density_spread_bound.py`.

| $x$ | spread$(x)$ |
|---|---|
| 0.0001 | 0.363403 |
| 0.01 | 0.365692 |
| 0.1 | 0.386307 |
| 0.5 | 0.472695 |
| 1.0 | 0.563738 |
| 1.5 | 0.629613 |
| 1.57 | 0.636544 |

Table 59.4: Direct check, at $K = 3, 5, 10$, of the collapsing-boundary construction from the proof of Theorem 59.13: interior boundaries $\phi_1, \phi_{K-1}$ placed symmetrically around a shrinking point $x_{\text{center}}$. $D$ stays above the floor and the excess shrinks with $x_{\text{center}}$ independently of $K$. Source: `code/density_spread_bound.py`.

| $K$ | $x_{\text{center}}$ | $D$ | excess over floor |
|---|---|---|---|
| 3 | 0.1 | 0.391196 | 0.027816 |
| 3 | 0.001 | 0.363650 | 0.000270 |
| 5 | 0.1 | 0.389242 | 0.025862 |
| 5 | 0.001 | 0.363635 | 0.000255 |
| 10 | 0.1 | 0.387775 | 0.024395 |
| 10 | 0.001 | 0.363623 | 0.000243 |

The floor value $1 - 2/\pi$ itself, and the reduction of every $K$ to the $K = 2$ function (Theorem 59.13's proof), are exact and closed-form: no numerical step enters them. Only the claim that spread$(x)$ has no interior dip below its two boundary values rests on a numerical certificate rather than a closed-form sign argument on spread$'(x)$. This is flagged explicitly, not smoothed over: the theorem as stated in Theorem 59.13 should be read as resting on Table 59.2's $2 \times 10^6$-point certificate for that one step, pending the closed-form argument requested below.

### 59.4.4 Where the natural schemes sit

Table 59.5 places the two schemes already studied in this monograph against the floor of Theorem 59.13. Neither comes close to it: both are comfortably-behaved schemes by construction (one holds angle constant, the other area), while the floor is approached only by a scheme that pushes both rings' widths toward zero at once, which neither natural scheme does.

Table 59.5: Density spread $D$ of the equiangular (Definition 1.1) and equal-area (Definition 59.1) schemes against the universal floor $1 - 2/\pi \approx 0.363380$, for six values of $K$. Both excesses grow with $K$: neither natural scheme approaches the floor as resolution increases; if anything they move further from it. Source: `code/density_spread_bound.py`, `data/density_spread_bound_results.json`.

| $K$ | $D$ (equiangular) | $D$ (equal-area) | excess (equiang.) | excess (eq.-area) |
|---|---|---|---|---|
| 2 | 0.527393 | 0.477465 | 0.164013 | 0.114085 |
| 4 | 0.780656 | 0.643484 | 0.417276 | 0.280104 |
| 8 | 0.895727 | 0.750036 | 0.532347 | 0.386656 |
| 16 | 0.949346 | 0.823501 | 0.585966 | 0.460120 |
| 32 | 0.975060 | 0.875164 | 0.611679 | 0.511784 |
| 64 | 0.987628 | 0.911686 | 0.624248 | 0.548306 |

**Key Result**

Theorem 59.13 answers item (i) of the closing todobox of an earlier draft of this chapter: no ring-partition scheme, of any order $K$, can bring its equatorial and polar-ring mean densities closer together than $1 - 2/\pi \approx 0.363$, and this bound is exact and uniform in $K$ — not merely observed for the equiangular and equal-area schemes, both of which in fact sit far above it (Table 59.5). The proof reduces the $K$-dependent, multi-boundary problem to the single-boundary $K = 2$ case *exactly* (Theorem 59.13's proof), via the closed-form monotonicity of Lemma 59.12; only the final one-variable floor rests on the numerical certificate of Table 59.2 rather than a closed-form derivative sign argument, as flagged above.

**To Complete / Verify**

Two items remain open from this chapter. (i′) A closed-form sign argument for $\text{spread}'(x) > 0$ (or, more directly, for $\text{spread}(x) > 1 - 2/\pi$) on the full open interval $(0, \pi/2)$, replacing the $2 \times 10^6$-point numerical certificate of Table 59.2 with an algebraic proof; the derivative has an exact closed form (given in Section 59.4.3) but no elementary sign argument for it has yet been found. (ii) An extension from one-dimensional ring partitions to genuine two-dimensional cell tessellations of the kind used by real DGGS (H3, S2, rHEALPix, DGGRID, OpenEAGGR), where "equal angle" and "equal area" are no longer the only two natural conventions, additional shape-regularity criteria (Kmoch et al. (2022)'s compactness measure, for instance) enter the picture, and a density-spread-type floor has not yet even been conjectured in closed form. Both are left open for future work.

Chapter 60

# Beyond the Equiangular Graticule: Boundary Rates Under General Ring Placement

## 60.1 The question left open in Chapter 2

Section 58.5 closed Chapter 58 with an explicit todobox: Proposition 58.1's boundary law $\rho_{\min}(K) = \tan(\pi/4K)$ was derived only for the equiangular graticule of Definition 1.1, and it was left open whether, or how, the $\Theta(K^{-1})$ boundary rate changes under a general (non-equiangular) ring placement — one where the boundaries $\phi_1, \dots, \phi_{K-1}$ are allowed to cluster near the pole rather than being spaced at constant angular increments. This chapter answers that question for an explicit, tractable one-parameter family of placements, within the general ring-partition framework of Definition 59.5 already introduced in Chapter 59. The answer is not merely a modified constant: it is a genuine change in the *qualitative* behaviour of the boundary ring, including a sharp transition at a specific, exactly identified parameter value.

## 60.2 A one-parameter family of ring placements

**Definition 60.1** (Power-law ring placement)**.** For $K \geq 2$ and $\beta > 0$, the *power-law scheme of exponent* $\beta$ is the ring-partition scheme (Definition 59.5) with boundaries

$$\phi_i(K, \beta) \;=\; \frac{\pi}{2}\left[1 - \left(1 - \frac{i}{K}\right)^{\beta}\right], \qquad i = 0, \dots, K. \tag{60.1}$$

*Remark* 60.2 (Well-posedness and the equiangular case). Writing $u = i/K \in [0, 1]$ and $f_\beta(u) := 1 - (1-u)^\beta$, $f_\beta$ is strictly increasing on $[0, 1]$ for every $\beta > 0$ (its derivative $\beta(1-u)^{\beta-1}$ is strictly positive on $[0, 1)$), with $f_\beta(0) = 0$ and $f_\beta(1) = 1$; hence Eq. (60.1) is a valid instance of Definition 59.5 for every $\beta > 0$. At $\beta = 1$, $f_1(u) = u$ exactly, so $\phi_i(K, 1) = \frac{\pi}{2} \cdot \frac{i}{K} = i\Delta\phi$: the power-law scheme reduces *exactly* to the equiangular graticule of Definition 1.1, not merely in some limit. This makes $\beta$ a genuine one-parameter deformation of the equiangular case, not an unrelated construction.

The area sequence of Definition 59.5, $a_i = \sin\phi_{i+1} - \sin\phi_i$, sums to 1 over $i = 0, \dots, K-1$ regardless of $\beta$ (telescoping $\sin(\pi/2) - \sin(0) = 1$); as in Chapter 1, write $\rho_i := a_i/a_0$ for the ring area normalized against the equatorial ring.

## 60.3 Exact closed form for the boundary ratio

**Proposition 60.3** (Exact boundary ratio under power-law placement)**.** *Under Definition 60.1,*

$$\rho_{K-1}(K,\beta) \;=\; \frac{a_{K-1}}{a_0} \;=\; \frac{1-\cos\!\left(\frac{\pi}{2}K^{-\beta}\right)}{\sin(\phi_1(K,\beta))}, \qquad \phi_1(K,\beta)=\frac{\pi}{2}\left[1-\left(1-\frac{1}{K}\right)^{\beta}\right], \tag{60.2}$$

*exactly, at every finite $K \geq 2$ and $\beta > 0$.*

*Proof.* By Definition 60.1, $\phi_0 = 0$, so $a_0 = \sin\phi_1 - \sin 0 = \sin\phi_1$, giving the denominator directly. For the numerator, $\phi_K = \pi/2$ and $\phi_{K-1} = \frac{\pi}{2}\left[1-(1/K)^{\beta}\right] = \frac{\pi}{2} - \frac{\pi}{2}K^{-\beta}$, so $a_{K-1} = \sin(\pi/2) - \sin(\phi_{K-1}) = 1 - \sin\!\left(\frac{\pi}{2} - \frac{\pi}{2}K^{-\beta}\right) = 1 - \cos\!\left(\frac{\pi}{2}K^{-\beta}\right)$, using $\sin(\pi/2 - x) = \cos x$. Dividing by $a_0$ gives Eq. (60.2). □

*Remark* 60.4 (Exact, not asymptotic, recovery of Proposition 58.1). At $\beta = 1$, $\phi_1(K,1) = \Delta\phi$ and $\phi_{K-1}(K,1) = \pi/2 - \Delta\phi$, so Eq. (60.2) reads $\rho_{K-1}(K,1) = [1-\cos\Delta\phi]/\sin\Delta\phi$, which is the standard half-angle identity for $\tan(\Delta\phi/2)$. This recovers Proposition 58.1's $\rho_{\min}(K) = \tan(\pi/4K)$ *exactly* at every finite $K$, confirmed independently in Table 60.1 to machine precision — not merely as the $\beta \to 1$ endpoint of an approximate family, which would be a much weaker check.

Table 60.1: Proposition 60.3 at $\beta = 1$ against Proposition 58.1's closed form $\tan(\pi/4K)$, for six values of $K$. Agreement to double-precision rounding confirms Eq. (60.2) is an honest generalization, not a reparametrization that merely resembles the original at $\beta = 1$. Source: `code/general_ring_placement_check.py`, `data/general_ring_placement_results.json`.

| $K$ | $\rho_{K-1}(K,1)$ | $\tan(\pi/4K)$ | $\lvert\Delta\rvert$ |
|---|---|---|---|
| 8 | $9.8491 \times 10^{-2}$ | $9.8491 \times 10^{-2}$ | 0 |
| 32 | $2.4549 \times 10^{-2}$ | $2.4549 \times 10^{-2}$ | $2.2 \times 10^{-16}$ |
| 128 | $6.1360 \times 10^{-3}$ | $6.1360 \times 10^{-3}$ | 0 |
| 512 | $1.5340 \times 10^{-3}$ | $1.5340 \times 10^{-3}$ | $2.2 \times 10^{-16}$ |
| 1024 | $7.6699 \times 10^{-4}$ | $7.6699 \times 10^{-4}$ | 0 |

Table 60.2 confirms Eq. (60.2) against a fully independent, direct computation of $a_0$ and $a_{K-1}$ from Definition 59.5 (implemented via the numerically stable Werner identity $\sin B - \sin A = 2\cos(\frac{A+B}{2})\sin(\frac{B-A}{2})$, the same identity used in the proof of Proposition 1.2, to avoid the catastrophic cancellation that a naive $1 - \sin\phi_{K-1}$ evaluation suffers once $\phi_{K-1}$ is close to $\pi/2$).

Table 60.2: Equation (60.2) (closed) against direct computation from Definition 59.5 (direct), for a representative grid of $(\beta, K)$. Source: `code/general_ring_placement_check.py`. Maximum absolute difference over the full grid tested ($\beta \in \{0.25, 0.5, 0.75, 1, 1.5, 2, 3\}$, $K \in \{4, \ldots, 128\}$): $7.1 \times 10^{-15}$.

| $\beta$ | $K$ | $\rho_{K-1}$ (direct) | $\rho_{K-1}$ (closed) | $\lvert\Delta\rvert$ |
|---|---|---|---|---|
| 0.25 | 32 | 16.933194 | 16.933194 | $7.1 \times 10^{-15}$ |
| 0.5 | 32 | 1.548597 | 1.548597 | $4.4 \times 10^{-16}$ |
| 0.75 | 32 | 0.184223 | 0.184223 | $7.2 \times 10^{-16}$ |
| 1.0 | 32 | 0.024549 | 0.024549 | $6.9 \times 10^{-18}$ |
| 2.0 | 32 | $1.2193 \times 10^{-5}$ | $1.2193 \times 10^{-5}$ | $1.8 \times 10^{-18}$ |

## 60.4 The asymptotic regime and a phase transition at $\beta = 1/2$

**Proposition 60.5** (Asymptotic boundary rate)**.** *For fixed $\beta > 0$, as $K \to \infty$,*

$$\rho_{K-1}(K,\beta) \;=\; \frac{\pi}{4\beta}\, K^{1-2\beta}\,\big(1+o(1)\big). \tag{60.3}$$

*Consequently $\rho_{K-1}(K,\beta) \to 0$ if $\beta > 1/2$, $\rho_{K-1}(K,\beta) \to \pi/2$ if $\beta = 1/2$, and $\rho_{K-1}(K,\beta) \to \infty$ if $\beta < 1/2$.*

*Proof.* Write $x_K := \frac{\pi}{2}K^{-\beta} \to 0$. By Taylor's theorem with Lagrange remainder, $1-\cos x = \frac{x^2}{2} + O(x^4)$, so the numerator of Eq. (60.2) is $\frac{1}{2}x_K^2(1+O(x_K^2)) = \frac{\pi^2}{8}K^{-2\beta}(1+O(K^{-2\beta}))$.

For the denominator, $(1-1/K)^\beta = 1 - \beta/K + O(K^{-2})$ by the binomial (Taylor) expansion of $(1-t)^\beta$ at $t=0$, valid for every fixed $\beta > 0$ and $K$ large enough that $1/K < 1$; hence $\phi_1(K,\beta) = \frac{\pi}{2}\big[\beta/K + O(K^{-2})\big] = \frac{\pi\beta}{2K} + O(K^{-2})$, and since $\sin y = y + O(y^3)$, $\sin\phi_1(K,\beta) = \frac{\pi\beta}{2K}\big(1+O(K^{-1})\big)$.

Dividing,

$$\rho_{K-1}(K,\beta) = \frac{\frac{\pi^2}{8}K^{-2\beta}(1+O(K^{-2\beta}))}{\frac{\pi\beta}{2K}(1+O(K^{-1}))} = \frac{\pi}{4\beta}K^{1-2\beta}\big(1+O(K^{-\min(1,2\beta)})\big),$$

which is Eq. (60.3). The three limiting behaviours follow immediately from the sign of the exponent $1-2\beta$, and at $\beta = 1/2$ the exponent vanishes, leaving the constant $\pi/(4\cdot\frac{1}{2}) = \pi/2$. □

*Remark* 60.6 (Consistency with Chapter 58). At $\beta = 1$, Eq. (60.3) gives $\rho_{K-1}(K,1) \sim \frac{\pi}{4}K^{-1}$, matching Remark 58.2's leading-order expansion of $\tan(\pi/4K)$ exactly, as it must given Remark 60.4.

Table 60.3 confirms Eq. (60.3) directly: for each $\beta$, an empirical decay exponent is estimated by a least-squares log–log slope of $\rho_{K-1}(K,\beta)$ against $K$ over $K \in \{4096, 8192, 16384, 32768\}$, fixed *before* comparison against the theoretical exponent $1-2\beta$ and not tuned to match it.

Table 60.3: Empirical log–log decay exponent of $\rho_{K-1}(K,\beta)$ against the theoretical exponent $1-2\beta$ of Proposition 60.5, and the value of $\rho_{K-1}$ at the largest $K$ tested ($K = 32768$). At $\beta = 1/2$ the value converges to $\pi/2 \approx 1.570796$, not to 0 or $\infty$. Source: `code/general_ring_placement_check.py`, `data/general_ring_placement_results.json`.

| $\beta$ | empirical slope | $1-2\beta$ | $\lvert\Delta\rvert$ | $\rho_{K-1}$ at $K = 32768$ |
|---|---|---|---|---|
| 0.25 | 0.5010 | 0.5000 | $1.0 \times 10^{-3}$ | 568.04 |
| 0.40 | 0.2001 | 0.2000 | $1.3 \times 10^{-4}$ | 15.707 |
| 0.50 | 0.00005 | 0.0000 | $4.6 \times 10^{-5}$ | 1.570774 |
| 0.60 | −0.2000 | −0.2000 | $2.4 \times 10^{-5}$ | 0.163623 |
| 0.75 | −0.5000 | −0.5000 | $1.3 \times 10^{-5}$ | 0.005785 |
| 1.00 | −1.0000 | −1.0000 | $5.6 \times 10^{-9}$ | 0.000024 |
| 1.50 | −2.0000 | −2.0000 | $2.5 \times 10^{-5}$ | $\sim 10^{-7}$ |
| 2.00 | −3.0001 | −3.0000 | $5.1 \times 10^{-5}$ | $\sim 10^{-10}$ |

**Key Result**

Unlike Chapter 58's equiangular case, where the pole ring's degeneration rate $\Theta(K^{-1})$ was a fixed structural fact, under a general power-law ring placement that rate is a free parameter of the placement, and it is not merely rescaled by $\beta$: it changes *sign* at the

exact, closed-form threshold $\beta = 1/2$ (Proposition 60.5). Below the threshold, the ring nominally "adjacent to the pole" does not shrink relative to the equatorial ring at all — it grows without bound. The equiangular case sits comfortably inside the shrinking regime ($\beta = 1 > 1/2$), but nothing about the discrete graticule construction itself guarantees this; it is a consequence of the specific, constant-angular-increment choice.

## 60.5 Does the pole remain the extremal ring?

Proposition 58.1 additionally asserted that ring $K-1$ is the ring of *globally smallest* area among $i = 0, \ldots, K-1$, not merely that its size decays at a particular rate. Proposition 60.5 already shows this cannot survive uniformly in $\beta$: for $\beta < 1/2$, ring $K-1$ grows without bound relative to ring 0, so it cannot be the global minimum for $K$ large enough. Table 60.4 checks this directly (not merely as a corollary of the asymptotic argument, since the asymptotic result concerns only ring $K-1$ against ring 0, not against *every* ring $i$): for each $\beta$ tested, across seven values of $K$ from 4 to 256, whether $\arg\min_i a_i$ actually equals $K-1$.

Table 60.4: Fraction of $K \in \{4, 8, 16, 32, 64, 128, 256\}$ for which ring $K-1$ is the global-minimum-area ring, by $\beta$. The transition sits exactly at the analytically identified threshold $\beta = 1/2$: for every $\beta \leq 1/2$ tested, ring $K-1$ is *never* the minimum (the minimum is instead at $i = 0$, the equatorial ring, in every case checked); for every $\beta > 1/2$ tested, ring $K-1$ is *always* the minimum. Source: `code/general_ring_placement_check.py`, `data/general_ring_placement_results.json`.

| $\beta$ | ring $K-1$ is global min | where the min is instead |
|---|---|---|
| 0.10 | 0/7 | $i = 0$ (equatorial ring), all cases |
| 0.25 | 0/7 | $i = 0$, all cases |
| 0.40 | 0/7 | $i = 0$, all cases |
| 0.50 | 0/7 | $i = 0$, all cases |
| 0.60 | 7/7 | — |
| 0.75 | 7/7 | — |
| 1.00 | 7/7 | — |
| 1.50 | 7/7 | — |
| 2.00 | 7/7 | — |
| 3.00 | 7/7 | — |
| 5.00 | 7/7 | — |

**Methodological Note**

The transition observed in Table 60.4 sits at exactly $\beta = 1/2$ over the finite grid tested, matching Proposition 60.5's threshold. This is reported as an empirical observation over that grid, not as a proven theorem: a closed-form argument that $i = 0$ is the global minimum for every $\beta \leq 1/2$ and every $K \geq 2$ (rather than merely for the seven $K$ values checked) has not been constructed, and is listed as an open item below. The asymptotic argument of Proposition 60.5 rigorously implies only that ring $K-1$ *eventually* exceeds ring 0 for $\beta < 1/2$ as $K \to \infty$; it does not by itself establish global minimality at every finite $K$, nor pin down $\beta = 1/2$ itself (a genuinely borderline case where Eq. (60.3)'s correction terms, not just its leading order, would need to be controlled).

## 60.6 Scope and open extensions

What is exact and closed-form in this chapter: the boundary-ratio formula Eq. (60.2) (an algebraic identity, proved directly from Definition 59.5, verified to machine precision in Table 60.2) and its exact reduction to Proposition 58.1 at $\beta = 1$ (Remark 60.4 and Table 60.1). What is rigorous but asymptotic: Proposition 60.5's decay-rate theorem, proved via standard Taylor remainders, not merely fitted to data (Table 60.3 confirms rather than discovers the exponent). What is numerical and explicitly not yet a theorem: the global-minimality transition of Section 60.5, checked over a finite grid and not proved for all $K$.

**To Complete / Verify**

Three items remain open. (i) A closed-form proof of the minimality transition at $\beta = 1/2$ for all $K \geq 2$, not only the finite grid of Table 60.4. (ii) The genuinely two-dimensional extension — to real DGGS cell geometries (H3, S2, rHEALPix, DGGRID, OpenEAGGR) rather than one-dimensional colatitude rings — already flagged as open in Section 58.5 and in Chapter 59's closing todobox; the phase-transition phenomenon identified here raises the more specific question of whether an analogous critical clustering exponent exists for two-dimensional cell refinement schemes, which has not been examined even at the level of a conjecture. (iii) A physical or design interpretation of $\beta < 1/2$: whether any DGGS construction in use today places boundaries this way near a pole (deliberately or as an artefact), or whether the regime is of purely mathematical interest as the boundary case delimiting where the classical "poles are where cells are smallest" intuition, taken for granted since the Mercator argument of Section 1.2, stops holding. None of the three is attempted here.

This chapter's contribution is narrow and precisely bounded: it converts Section 58.5's qualitative open question — does the boundary rate depend on ring clustering? — into an exact, closed-form answer for one explicit family, and shows that the dependence is not merely quantitative but includes a genuine phase transition at an exactly identified threshold. It does not close Chapter 58's second open item (real DGGS geometry), which remains exactly as open as it was.

Chapter 61

# Why Real DGGS Don't Obey the Graticule's Boundary Law

## 61.1 Scope: the connection item, from two todoboxes at once

Two separate open items point at the same question. Section 58.5's item (ii) asked whether an actual DGGS shows the same kind of boundary statistic as the toy lat/lon graticule's exact $\rho_{\min}(K) = \tan(\pi/4K) = \Theta(K^{-1})$ law. Section 11.5's item (iii) asked the same thing from the data side: is there a quantitative relationship between that closed-form rate and the empirical area-variation figures of Chapters 11 and 13? Table 13.1 now makes the answer to both concrete, and it is not the one a naive analogy would predict.

## 61.2 The graticule shrinks toward the pole; H3 and S2 don't

Proposition 58.1 says the toy graticule's worst-case cell-area ratio $\rho_{\min}(K)$ *decays to zero* as resolution $K$ increases: refine the grid, and the pole cell's distortion shrinks, just more slowly than the typical cell's. Table 13.1 shows the opposite qualitative behaviour for both real DGGS studied: H3's normalized-area $\sigma$ is flat from the coarsest resolution tested onward ($0.13007$ at res. 2, still $0.13058$ at res. 4), and S2's, after an initial transient, converges to a nonzero constant near $0.152$ rather than to $0$. Refining an H3 or S2 tessellation does not make the normalized cell-area distribution more uniform in the limit; it reproduces essentially the same distribution at every resolution past the first few levels.

**Key Result**

This is not a contradiction between Chapter 58 and Chapter 13; it is a difference in what is being held fixed. The graticule of Definition 1.1 is built by dividing a *fixed* interval $[0, \pi/2]$ into $K$ equal rings: as $K$ grows, each individual ring covers a shrinking angular width, and the ring nearest the pole is squeezed by the sphere's curvature more than the equatorial ring is — a genuine boundary effect that vanishes, at rate $K^{-1}$, only because the rings keep shrinking. H3 and S2, by contrast, are built by *recursive subdivision*: each cell at resolution $K + 1$ is (to a first approximation, exactly for S2's quadtree, approximately for H3's aperture-7 scheme) a locally-rescaled copy of its resolution-$K$ parent. A recursively self-similar construction has no mechanism to make relative distortion shrink with resolution, because the map from "parent cell" to "normalized child-cell shapes" is (again, approximately) the same map at every resolution: whatever normalized-area spread exists among a cell's children at resolution $2$ already exists, essentially unchanged, among their children's children at resolution $20$.

## 61.3 The self-similarity argument is a sketch, not a proof

**To Complete / Verify**

The paragraph above is offered as the most economical explanation consistent with Table 13.1's numbers, not as something this volume has proved. Two gaps are explicit. First, H3's subdivision is only *approximately* self-similar: aperture-7 children are rotated and slightly displaced relative to a naive rescaling of the parent (this is exactly why H3 uses non-integer "aperture" language rather than a clean recursive tiling), and the twelve pentagon lineages break self-similarity outright at every resolution along their arm — consistent with Table 12.1's finding that the pentagons sit at a fixed normalized area ($\approx 0.51$) independent of resolution, which is what self-similarity predicts for them specifically, but the hexagon-vs-pentagon population mix at each resolution is not itself scale-invariant in an elementary sense that has been checked. Second, no attempt is made here to derive $0.152$ as S2's asymptotic normalized-area $\sigma$ from the geometry of the cube-to-sphere gnomonic projection that generates S2 cells; the saturation is documented empirically (Table 13.1) but not derived analytically, which a genuine closed-form boundary law, in the sense of Proposition 58.1, would require.

## 61.4 The bulk/extreme distinction, once more

One part of Chapter 58's structure *does* transfer cleanly, though in a different guise. Remark 13.1 found that S2's normalized-area standard deviation converges smoothly across resolution while its minimum converges noisily. This is the same bulk-versus-extreme distinction that separates Proposition 58.3 ($\Theta(K^{-2})$, a supremum bound that is nonetheless a smooth deterministic function of $K$) from Proposition 58.1 ($\Theta(K^{-1})$, attained at a single identified ring). The graticule case is fully deterministic — both rates are exact closed-form functions of $K$ — so "noise" is not the right word there; in the real-DGGS case, where a resolution's worst cell is effectively drawn from an evolving finite population rather than fixed by a formula, the same underlying distinction (an average over many cells vs. a statistic of the single worst one) manifests as smooth convergence vs. resolution-to-resolution jitter instead of as two different power laws. The mechanism connecting the two settings is the same; the observable symptom differs because one setting is a closed-form recursion and the other is an empirical population of cells.

## 61.5 What remains open

**To Complete / Verify**

This closes item (ii) of Section 58.5 and item (iii) of Section 11.5 in the qualified sense above: the question has a documented empirical answer (real DGGS are resolution-invariant rather than boundary-decaying, for a sketched self-similarity reason) rather than the closed-form quantitative relationship both todoboxes originally asked for. A genuine closed-form treatment would need to (a) make the self-similarity argument precise for H3's aperture-7 scheme including the pentagon lineages, and (b) derive S2's $\approx 0.152$ asymptote from its cube-face gnomonic projection rather than reading it off a table. Neither is attempted here.

Chapter 62

# From the Graticule to a Real Grid: Does the Boundary Law Survive Contact with H3?

## 62.1 Scope: the item ch07 flagged as not yet attempted

Section 114.5's ledger disclosed, among the volume's open items, that Section 58.5's non-equiangular generalization "has now been carried out for one explicit family (Chapter 60), but its application to a genuine DGGS cell geometry, rather than the toy lat/lon graticule, still has not been attempted." Proposition 58.1 and Proposition 60.3 are exact statements about an idealized ring partition of the sphere; nothing in their proofs guarantees that any statement about a real, independently constructed discrete global grid system follows from them. This chapter makes a first, fully disclosed attempt to find out, using H3 (the same grid already given from-scratch, independently verified area treatment in Chapter 12) as the test case.

## 62.2 What is actually being compared

H3 is not built from ring boundaries at all: it is a quasi-uniform hexagonal (and twelve-pentagon) tessellation derived from icosahedral subdivision, and no step of its construction refers to the equiangular graticule of Definition 1.1. Any comparison therefore has to define, independently of H3's own construction, a way of assigning real H3 cells to the graticule's rings. This chapter uses the simplest possible rule, stated in full: for a chosen ring count $K$, the same boundaries $\phi_i = i \cdot (\pi/2)/K$, $i = 0, \ldots, K$ used throughout Chapter 58 partition $[0, \pi/2]$ into $K$ latitude bands; every real H3 cell, at a chosen resolution, is assigned to the band containing its own centroid latitude $|\mathrm{lat}(c)|$ (both hemispheres pooled, since the boundary law is symmetric in latitude). Each band's real area $A_i = \sum_{c\,:\,\mathrm{band}(c)=i} \texttt{h3.cell_area}(c)$ uses exactly the same `h3.cell_area()` call already cross-checked against H3's own reported average cell area in Chapter 12, not any hand-rolled or idealized formula. The empirical ratio $\rho_i := A_i/A_0$ is then compared against Eq. (60.2)'s $\rho_{K-1}(K, \beta)$ for a scan of $\beta$ values, and specifically against $\beta = 1$'s closed form $\tan(\pi/4K)$ from Proposition 58.1.

**Methodological Note**

This is a centroid-binning rule chosen for its simplicity, not because it is the only or the most natural way to compare a hexagonal tessellation with a ring partition. A hexagonal cell whose centroid falls just inside band $i$ may have a third of its own area lying physically in band $i-1$ or $i+1$; the rule assigns all of that cell's area to band $i$ regardless. This is a real source of mismatch between $A_i$ and the true physical band area, disclosed here rather than smoothed over, and is discussed further in Section 62.4 as part of why the agreement found below should not be over-read.

## 62.3 Results: three resolutions, three ring counts

Table 62.1 reports the full run at three H3 resolutions, with $K$ chosen at each resolution so that even the polar-most band retains a non-trivial cell count (disclosed in the table rather than assumed adequate).

Table 62.1: Real H3 cell-area mass, binned into $K$ equiangular latitude bands using Definition 1.1's own boundaries, against Proposition 58.1's $\beta = 1$ prediction for the polar-most band's ratio $\rho_{K-1}$. $n_{K-1}$ is the real cell count landing in that polar-most band (never zero, so the ratio is never computed from an empty band). Source: `code/h3_ring_placement_law_check.py`, `data/h3_ring_placement_law_check_results.json`.

| res | $K$ | cells (total) | $n_{K-1}$ | $\rho_{K-1}$ (real) | $\tan(\pi/4K)$ | ratio |
|---|---|---|---|---|---|---|
| 2 | 6 | 5,882 | 184 | 0.127457 | 0.131652 | 0.9681 |
| 3 | 10 | 41,162 | 478 | 0.078965 | 0.078702 | 1.0033 |
| 4 | 16 | 288,122 | 1,298 | 0.049549 | 0.049127 | 1.0086 |

Across all three configurations the real-to-theory ratio for the polar-most band stays within 3.2% of unity, and within 0.9% at the two finer resolutions. This is not, by itself, evidence that H3 "obeys" the graticule's boundary law: Section 62.3.1 checks whether this closeness is specific to $\beta = 1$ or would appear for any exponent in the family.

### 62.3.1 Is the match specific to $\beta = 1$?

For each configuration, Eq. (60.2)'s $\rho_{K-1}(K, \beta)$ was evaluated over $\beta \in \{0.25, 0.5, 0.75, 1, 1.25, 1.5, 2, 3\}$ and compared against the same real $\rho_{K-1}$ figure. At all three resolutions the minimum-absolute-difference $\beta$ over this scan is exactly $\beta = 1$ (absolute differences $4.2 \times 10^{-3}$, $2.6 \times 10^{-4}$, and $4.2 \times 10^{-4}$ respectively, against differences an order of magnitude or more larger for the next-closest scanned $\beta$ values, 0.75 and 1.25, at every resolution). The match is therefore not an artefact of any normalized ratio necessarily landing near the real figure for a suitably chosen $\beta$: the specific equiangular exponent that Chapter 58 originally derived for the idealized graticule is also the specific exponent that best predicts real H3 polar-band mass, among the family Chapter 60 later generalized to.

### 62.3.2 Why the profile shape itself is not read the same way

A second, weaker diagnostic was attempted: fitting the log-log slope of $\rho_i$ against band index $i$ (bands $i = 1, \ldots, K-1$, at fixed $K$) and comparing it to Proposition 60.5's $1 - 2\beta$. Unlike Table 62.1's single-band comparison, this fits every band simultaneously rather than isolating the asymptotic, polar-most regime the original exponent check in Chapter 60 deliberately restricted itself to (that check used only the four largest-$K$ points for exactly this reason). The resulting slopes ($-1.05$, $-0.87$, $-0.77$ at res 2, 3, 4) drift away from $\beta = 1$'s theoretical $-1$ as resolution and $K$ grow, in the opposite direction from what Section 62.3.1 found for the polar-most band alone. This is reported as an open, unreconciled secondary finding rather than folded into Section 62.3.1's cleaner result: the two diagnostics are testing different things (one extreme ratio at fixed $K$, versus a whole radial profile's shape at fixed $K$), and no claim is made here about why they disagree in trend.

## 62.4 An honest candidate explanation, not a proof

Archimedes' hatbox theorem gives the true physical area of the spherical band between $\phi_i$ and $\phi_{i+1}$ exactly, as a function of those two latitudes alone, regardless of what lies inside the band. If H3's real cells were assigned to bands by their true areal overlap (splitting each boundary-straddling cell's area between the bands it actually occupies) rather than by centroid membership, $A_i$ would equal the idealized $a_i$ exactly, for any tessellation whatsoever, and Table 62.1's agreement would be a triviality of the hatbox theorem rather than a fact about H3. Centroid binning does not do this: a cell straddling a band boundary has its whole area credited to one side. The plausible reason Table 62.1 still shows close agreement is that, at each resolution tested, boundary-straddling cells are numerous and small relative to each band, so that area credited wrongly to band $i$ instead of $i \pm 1$ is approximately balanced, in aggregate, by area wrongly credited the other way – a statistical-cancellation argument, not a derivation. This chapter does not prove this cancellation claim; it is offered as the most likely explanation consistent with Section 62.3.1's finding that the agreement is specific to $\beta = 1$ (a generic cancellation argument alone would not obviously favour one $\beta$ over another) and is left as an explicit target for a future chapter rather than asserted here as settled.

**Key Result**

Real H3 cell-area mass, binned by centroid latitude into the same equiangular rings Chapter 58 defined for the idealized graticule, matches Proposition 58.1's $\beta = 1$ boundary prediction to within 3.2% at three independent resolutions, and $\beta = 1$ is the specific best fit among eight scanned exponents at all three – not merely one exponent among several giving comparable agreement. This closes Section 114.5's "still has not been attempted" item for a from-scratch, real DGGS grid, with an outcome (partial, quantified agreement, not a clean confirmation) that is itself disclosed as only partially understood: a candidate explanation is offered in Section 62.4 without being proved, and a second diagnostic (Section 62.3.2) shows the opposite trend without being reconciled.

**To Complete / Verify**

Three items are left open by this chapter. **(i)** The statistical-cancellation explanation in Section 62.4 is not derived; a genuine derivation would need to relate H3's cell-size distribution near a given latitude to the centroid-binning error it produces, which this chapter has not attempted. **(ii)** The opposite trend found by the log-log slope diagnostic (Section 62.3.2) against the clean single-band result (Section 62.3.1) is reported but not reconciled. **(iii)** This chapter tests only H3; whether the same $\beta = 1$-specific agreement appears for S2 or rHEALPix, both already given independent from-scratch geometric treatment elsewhere in this volume (Chapters 95 and 97), is left for a future pass using the code already built for those chapters.

Chapter 63

# Three Grids, One Ring Law: Extending the Check to S2 and rHEALPix

## 63.1 Scope: closing item (iii) of ch:h3-ring-placement-law's todobox

Chapter 62's closing todobox (Section 62.4) left open, as its third item, "whether the same $\beta = 1$-specific agreement appears for S2 or rHEALPix, both already given independent from-scratch geometric treatment elsewhere in this volume (Chapters 95 and 97), ... using the code already built for those chapters." This chapter runs exactly that check, reusing each grid's own real geometry engine (`s2sphere`'s `exact_area()` and `get_center()`; `rhealpixdggs`'s `area(plane=False)` and `nucleus(plane=False)`) and the identical centroid-binning method and $\beta$-scan as Chapter 62's H3 check, changing nothing about the method itself.

## 63.2 Results

Table 63.1: The polar-most-band ratio check of Chapter 62, run identically on all three grids this volume has given independent real-geometry treatment. H3's row reproduces Table 62.1. Source: `code/s2_ring_placement_law_check.py`, `code/rhealpix_ring_placement_law_check.py`, `data/s2_ring_placement_law_check_results.json`, `data/rhealpix_ring_placement_law_check_results.json`.

| grid | res/level | $K$ | cells | $\rho_{K-1}$ (real) | $\tan(\pi/4K)$ | ratio |
|---|---|---|---|---|---|---|
| H3 | 2 | 6 | 5,882 | 0.127457 | 0.131652 | 0.9681 |
| H3 | 3 | 10 | 41,162 | 0.078965 | 0.078702 | 1.0033 |
| H3 | 4 | 16 | 288,122 | 0.049549 | 0.049127 | 1.0086 |
| S2 | 4 | 6 | 1,536 | 0.123241 | 0.131652 | 0.9361 |
| S2 | 5 | 10 | 6,144 | 0.065368 | 0.078702 | 0.8306 |
| S2 | 6 | 16 | 24,576 | 0.047030 | 0.049127 | 0.9573 |
| rHEALPix | 2 | 6 | 486 | 0.166667 | 0.131652 | 1.2660 |
| rHEALPix | 3 | 10 | 4,374 | 0.066138 | 0.078702 | 0.8404 |
| rHEALPix | 4 | 16 | 39,366 | 0.045455 | 0.049127 | 0.9252 |

Three things are immediately visible in Table 63.1 that were not visible from H3 alone.

**First**, H3's agreement is not typical of DGGS in general: S2's deviation from unity ranges from 4.3% to 16.9% across the three configurations tested, and rHEALPix's from 7.5% to 26.6% – both substantially larger than H3's 0.9–3.2% range. Whatever mechanism produces H3's close agreement (Section 62.4's candidate statistical-cancellation explanation, offered there without proof) is evidently not shared, or not shared to the same degree, by S2 or rHEALPix.

**Second**, the sign of the deviation is not consistent across grids. H3's ratio sits close to but on both sides of 1 (below at the coarsest configuration, above at the two finer ones). S2's ratio is

below 1 at all three configurations tested (real polar mass falls short of the graticule prediction). rHEALPix's ratio is above 1 at the coarsest configuration and below 1 at the two finer ones – the opposite coarse-resolution sign from H3. No single directional bias (consistently "too much" or "too little" polar mass relative to the idealized ring law) holds across all three real grids.

**Third**, despite this, the beta-scan finding from Section 62.3.1 reproduces exactly on both S2 and rHEALPix: at every one of the six new (grid, configuration) pairs, $\beta = 1$ is the specific minimum-absolute-difference exponent among the eight scanned values, exactly as it was for all three H3 configurations. This is the more robust of the two findings this volume can now report: not that any of the three real grids' polar-band mass matches the idealized ring law closely in absolute terms, but that whichever exponent in the Chapter 60 family the real data comes closest to is consistently the equiangular one, across three geometrically unrelated grid constructions (icosahedral-hexagonal, cubed-sphere quadrilateral, and HEALPix-derived diamond/square).

**Key Result**

Item (iii) of Chapter 62's todobox is closed: the $\beta = 1$-specific best fit does appear for both S2 and rHEALPix, at all three configurations tested for each, exactly as it did for H3. The closeness of the absolute match to the idealized law, however, is specific to H3 and does not generalize – S2 and rHEALPix show substantially larger and sign-inconsistent deviations from $\tan(\pi/4K)$ itself. The volume's overall real-DGGS finding is therefore narrower than a first reading of Chapter 62 alone might suggest: what survives contact with three real grids is the identity of the best-fitting exponent, not the numerical closeness of the fit.

**To Complete / Verify**

Chapter 62's item (iii) is closed by this chapter; its items (i) and (ii) remain exactly as that chapter left them, now joined by two more: **(iv)** why H3's absolute agreement is closer than S2's or rHEALPix's, and **(v)** why the sign of the deviation differs by grid and by resolution rather than following a single consistent bias, are both left open. A derivation connecting each grid's own cell-size and cell-shape distribution near a given latitude band to the centroid-binning error it produces – the mechanism Section 62.4 proposed without proving for H3 alone – would need to explain all three grids' figures simultaneously to be considered a genuine account rather than a post hoc fit to any one of them.

Chapter 64

# Testing a Candidate Mechanism: Does Within-Band Heterogeneity Explain the Cross-Grid Gap?

## 64.1 Scope: items (iv) and (v) of ch:ring-placement-law-s2-rhealpix's todobox

Chapter 63's closing todobox (Section 63.2) left open two items once the $\beta = 1$-specific best fit had been confirmed across all three real grids: **(iv)** why H3's absolute agreement with $\tan(\pi/4K)$ is closer than S2's or rHEALPix's, and **(v)** why the sign of the deviation differs by grid and by resolution rather than following one consistent bias. That chapter proposed, without proof, that the mechanism might lie in each grid's own cell-size and cell-shape distribution near the band boundary. This chapter turns that proposal into a specific, falsifiable test.

**Methodological Note**

The hypothesis tested here is deliberately narrow and stated in advance of computing anything: *larger heterogeneity in the individual real cell areas making up the polar-most band* (*band* $K-1$) *correlates with a larger* $|\text{ratio} - 1|$ *deviation from the idealized law*. Heterogeneity is operationalized as the coefficient of variation, $\text{CV} = \sigma/\mu$, of that band's individual cell areas – not their sum, which is all the three source chapters recorded. This is one specific candidate mechanism among possibly several; a negative result here rules out this mechanism, not the existence of some other one.

## 64.2 Method

The nine (grid, configuration) pairs of Table 63.1 are regenerated from scratch, reusing exactly the same real geometry engines and band-assignment rule as Chapters 62 and 63 (`h3-py`'s `cell_area()/cell_to_latlng()`; `s2sphere`'s `exact_area()/get_center()`; `rhealpixdggs`'s `area(plane=False)/ nucleus(plane=False)`). The only change is that individual cell areas within band $K-1$ are now retained as a list rather than summed immediately, so that their coefficient of variation $\text{CV}_{K-1}$ can be computed. As an internal consistency check, the ratio $\rho^{\text{real}}_{K-1}/\tan(\pi/4K)$ is also recomputed independently here and compared against Table 63.1's already-published values. Source: `code/ring_placement_bandmass_heterogeneity_check.py`, `data/ring_placement_bandmass_heterogeneity_results.json`.

Table 64.1: Coefficient of variation of individual cell areas within the polar-most band $(K-1)$, against the already-published absolute deviation from the idealized law. The ratio column is recomputed independently from scratch here and matches Table 63.1 to the precision shown, an internal reproducibility check on that earlier chapter.

| grid | config | $n_{K-1}$ | ratio (recomputed) | $\lvert\text{ratio}-1\rvert$ | $\text{CV}_{K-1}$ |
|---|---|---|---|---|---|
| H3 | res 2, $K=6$ | 184 | 0.9681 | 0.0319 | 0.0864 |
| H3 | res 3, $K=10$ | 478 | 1.0033 | 0.0033 | 0.0632 |
| H3 | res 4, $K=16$ | 1,298 | 1.0086 | 0.0086 | 0.0401 |
| S2 | lvl 4, $K=6$ | 48 | 0.9361 | 0.0639 | 0.0533 |
| S2 | lvl 5, $K=10$ | 64 | 0.8306 | 0.1694 | 0.0476 |
| S2 | lvl 6, $K=16$ | 120 | 0.9573 | 0.0427 | 0.0382 |
| rHEALPix | res 2, $K=6$ | 18 | 1.2660 | 0.2660 | 0.0000 |
| rHEALPix | res 3, $K=10$ | 50 | 0.8404 | 0.1596 | 0.0000 |
| rHEALPix | res 4, $K=16$ | 162 | 0.9252 | 0.0748 | 0.0000 |

## 64.3 Results

The recomputed ratios reproduce Table 63.1 exactly, confirming that this chapter's independent regeneration of all nine configurations is consistent with the earlier one.

**The heterogeneity hypothesis fails, in a specific and informative way.** The Pearson correlation between $\text{CV}_{K-1}$ and $\lvert\text{ratio}-1\rvert$ across all nine points is $r=-0.619$: negative, and therefore opposite in sign to the hypothesis, which predicted a positive relationship. The clearest single counterexample is rHEALPix: its polar-band cells have *exactly zero* area variation at all three resolutions tested ($\text{CV}_{K-1}=0$ to machine precision), yet rHEALPix shows the *largest* deviations of all nine (grid, configuration) points tested, including this chapter's largest value, $0.2660$, at resolution 2.

**Key Result**

The within-band cell-area heterogeneity mechanism proposed in Chapter 63 is ruled out: the correlation across all three grids runs in the wrong direction ($r=-0.619$), and rHEALPix's exactly-zero heterogeneity coexists with its largest deviations. This is not a coincidental floor effect – rHEALPix's construction is authalic (area-preserving) by design at every resolution, so its cells within any single equiangular band are expected to carry near-identical area regardless of how close or far that band's summed mass sits from the idealized graticule law. The two phenomena (equal-area construction; agreement with the equiangular ring law) are logically independent properties, and this chapter's data confirms they are empirically independent too.

## 64.4 What this rules in, and what remains open

Ruling out one candidate mechanism is itself a result, in the sense Chapter 114 uses throughout this volume: it narrows, rather than closes, items (iv) and (v). What this chapter's negative result does suggest is that the explanation is unlikely to be a property of the *area* distribution within the band at all, since the grid with zero area variance (rHEALPix) is not the grid with the closest agreement (H3 is). This points instead toward *shape*, not area, as the more promising direction: rHEALPix's diamond/square cells and S2's cubed-sphere quadrilaterals both undergo more severe angular distortion near the pole than H3's quasi-regular hexagons, and

this volume already has the instrument to measure that directly, the isoperimetric quotient (IPQ) used in Chapters 100 and 102, but has not yet applied it band-by-band to this specific ring-placement question.

**To Complete / Verify**

Item (iv) (why H3 agrees more closely) and item (v) (why the deviation's sign differs by grid) both remain open after this chapter closes the area-heterogeneity candidate. The natural next test, not yet attempted, is whether polar-band IPQ (rather than area CV) correlates with $|\text{ratio} - 1|$ across the same nine points, reusing Chapters 100 and 102's own IPQ computation restricted to each band's cells. A second, independent possibility not yet tested at all: the deviation may reflect not cell area or shape but the graticule band-boundary rule itself interacting with each grid's discrete cell lattice differently (an aliasing-like effect between the continuous boundary $\phi_i = i(\pi/2)/K$ and each grid's own discrete cell edges), which would require comparing results under a small perturbation of the band boundary rather than any property of the cells themselves.

Chapter 65

# Shape, Not Area: Testing the Polar-Band IPQ Correlation

## 65.1 Scope: the second candidate mechanism from Chapter 64's todobox

Chapter 64 ruled out within-band cell-area heterogeneity as the mechanism behind items (iv) and (v) of Chapter 63's closing todobox, and pointed instead toward cell *shape*: rHEALPix's zero area variance coexists with its largest deviations, while H3's quasi-regular hexagons show the closest agreement with the idealized law. That chapter's own todobox proposed, as the natural next test, checking whether polar-band isoperimetric quotient (IPQ) – the instrument already built and used elsewhere in this volume (Chapters 100 and 102) – correlates with $|\text{ratio} - 1|$ across the same nine (grid, configuration) points. This chapter runs that test.

**Methodological Note**

The hypothesis, stated in advance of computing anything: *lower mean IPQ in the polar-most band (band $K-1$) – i.e., cells that depart further from the isoperimetric ideal of a circle – correlates with larger* $|\text{ratio} - 1|$ *deviation from the idealized ring-placement law*. A negative Pearson correlation between IPQ and $|\text{ratio} - 1|$ would support this; a weak or positive one would rule it out, exactly as Chapter 64 treated its own CV hypothesis.

## 65.2 Method

The same nine (grid, configuration) pairs and the same band-$K-1$ cell sets are regenerated from scratch, reusing unchanged the real geometry engines and band-assignment rule of Chapters 62 to 64. For each cell in band $K-1$, $\text{IPQ} = 4\pi A/P^2$ is computed instead of area alone. S2 and rHEALPix reuse, unmodified, the refined-perimeter machinery already built for Chapters 100 and 102's own predecessor chapter (32 interpolated points per edge, mapped through each library's own forward/inverse projection rather than a naive 4-corner chord); rHEALPix's refinement is applied here on the same UNIT_SPHERE, $N_{\text{side}} = 3$ instantiation the ring-placement chapters use, not the WGS84-ellipsoid instantiation that earlier chapter used, since IPQ is scale-free and what must match here is the band-assignment geometry, not the ellipsoid. H3 has no prior IPQ chapter in this volume to reuse: its cells' boundary vertices (`h3.cell_to_boundary()`) are already the cell's true edges, with no plane/inverse-projection shortcut to bypass the way rHEALPix's quad/cap shapes have, so a geodesic-chord perimeter over those vertices is used directly rather than further refined. All three grids' areas are read from each library's own native steradian-scale call (`rads^2`, `exact_area()`, `area(plane=False)`); IPQ is scale-invariant under a single consistent radius choice, so the three grids' different internal conventions do not need to be reconciled to a common $R$. As an internal consistency check, the ratio $\rho^{\text{real}}_{K-1}/\tan(\pi/4K)$ is independently recomputed

here too and compared against Tables 63.1 and 64.1's already-published values. Source: `code/ipq_band_correlation_check.py`, `data/ipq_band_correlation_results.json`.

## 65.3 Results

Table 65.1: Mean, min, and max IPQ of band $K-1$'s individual real cells, against the already-published $|\text{ratio}-1|$ (recomputed independently here and matching Table 64.1 exactly, an internal reproducibility check).

| grid | config | $n_{K-1}$ | $\lvert\text{ratio}-1\rvert$ | IPQ mean | IPQ min | IPQ max |
|---|---|---|---|---|---|---|
| H3 | res 2, $K=6$ | 184 | 0.0319 | 0.9061 | 0.9014 | 0.9072 |
| H3 | res 3, $K=10$ | 478 | 0.0033 | 0.9062 | 0.9001 | 0.9098 |
| H3 | res 4, $K=16$ | 1,298 | 0.0086 | 0.9064 | 0.9055 | 0.9068 |
| S2 | lvl 4, $K=6$ | 48 | 0.0639 | 0.7836 | 0.7799 | 0.7864 |
| S2 | lvl 5, $K=10$ | 64 | 0.1694 | 0.7848 | 0.7834 | 0.7857 |
| S2 | lvl 6, $K=16$ | 120 | 0.0427 | 0.7850 | 0.7840 | 0.7855 |
| rHEALPix | res 2, $K=6$ | 18 | 0.2660 | 0.7321 | 0.6972 | 1.0023 |
| rHEALPix | res 3, $K=10$ | 50 | 0.1596 | 0.7109 | 0.6464 | 1.0004 |
| rHEALPix | res 4, $K=16$ | 162 | 0.0748 | 0.7039 | 0.6162 | 1.0002 |

The recomputed ratios reproduce Table 64.1 exactly at every one of the nine points, confirming this chapter's independent regeneration is consistent with both earlier chapters.

**The IPQ hypothesis is not ruled out, and points the same direction as the grid-level ordering already visible in Chapter 64's discussion.** The Pearson correlation between mean band-$K-1$ IPQ and $|\text{ratio}-1|$ across all nine points is $r=-0.686$: negative, as the hypothesis predicted, and considerably stronger than the $r=+0.619$ (wrong-signed) correlation the area-heterogeneity hypothesis produced. The three grids separate cleanly by IPQ level – H3 ($\approx 0.906$), S2 ($\approx 0.784$), rHEALPix ($\approx 0.70$–$0.73$) – in exactly the same order as their typical agreement with the idealized law (H3 closest, rHEALPix least close), matching Chapter 64's own qualitative discussion of shape as the more promising direction.

**Key Result**

Polar-band mean IPQ correlates negatively with $|\text{ratio}-1|$ across the nine (grid, configuration) points ($r=-0.686$), in the direction the shape hypothesis predicted and with a magnitude comparable to, but not stronger by an order of magnitude than, the area-CV hypothesis's own (wrong-signed) correlation. This is real support for shape over area as the more relevant property, and it explains item (iv) at the level of a grid-to-grid ordering (H3's more isoperimetric cells track its closer agreement).

## 65.4 What the correlation does, and does not, explain

The correlation is driven almost entirely by the between-grid spread, not by within-grid resolution trends. Within each single grid, mean band-$K-1$ IPQ is nearly flat across the three resolutions tested (H3: $0.9061 \to 0.9062 \to 0.9064$; S2: $0.7836 \to 0.7848 \to 0.7850$; rHEALPix: $0.7321 \to 0.7109 \to 0.7039$, the one grid with a visible, if still small, downward drift) while $|\text{ratio}-1|$ is not monotonic at all within any of the three grids (e.g. S2's own deviation runs $0.0639 \to 0.1694 \to 0.0427$, non-monotonic in $K$). A single scalar (mean IPQ of the band)

that barely moves with resolution cannot, by itself, be the mechanism behind a deviation that swings both up and down as resolution changes within one grid – so this chapter's positive-ish result answers item (iv) (why H3 agrees more closely, in a cross-grid sense) more than item (v) (why the sign flips resolution-by-resolution within a grid). rHEALPix's own band also shows the widest internal IPQ spread of the three grids at every resolution (min as low as $0.616$ against a max of essentially $1.000$ at the same resolution), consistent with that band mixing authalic cap cells (near-circular, IPQ close to 1) with skew-quad cells (far from circular) – a within-band heterogeneity of *shape*, distinct from the area heterogeneity Chapter 64 already measured and ruled out for the same cells.

**To Complete / Verify**

This chapter narrows, rather than closes, items (iv) and (v): the cross-grid IPQ ordering is a real, disclosed piece of the (iv) puzzle, but the near-flatness of within-grid IPQ against a non-monotonic within-grid deviation means item (v)'s resolution-by- resolution sign flip is not explained by this band's mean shape either. Two directions remain open. First, the aliasing-like mechanism Chapter 64 already proposed as a second, independent candidate – interaction between the continuous graticule boundary $\phi_i = i(\pi/2)/K$ and each grid's own discrete cell lattice – is still entirely untested. Second, this chapter used the band's *mean* IPQ; whether the band's IPQ *variance* (as opposed to its area-CV, already ruled out) tracks the within-grid resolution-to-resolution sign changes is a distinct, narrower test not attempted here.

Chapter 66

# The Boundary Itself, Not the Cells: A Third Candidate Mechanism

## 66.1 Scope: the second untested direction from Chapter 64's todobox

Chapter 64 ruled out within-band area heterogeneity as the mechanism behind items (iv) and (v) of Chapter 63's closing todobox, and Chapter 65 found the band's mean isoperimetric quotient (IPQ) explains item (iv) but not item (v). Both chapters tested a property *of the cells* inside the polar-most band. A second, logically independent candidate was left entirely untested by both: that the deviation reflects not any property of the cells but an interaction between the graticule band boundary $\phi_i = i(\pi/2)/K$ itself – an arbitrary, continuous choice – and each grid's fixed, discrete cell lattice, the same kind of aliasing-like effect that arises whenever a continuous partition is imposed on a discrete signal. This chapter tests that candidate directly.

**Methodological Note**

**What was done, precisely.** The nine (grid, configuration) points of Table 63.1 are regenerated from scratch, reusing unchanged the real geometry engines and per-cell latitude/area calls of Chapters 62 to 64. The only change is that every band boundary is shifted uniformly by $\varepsilon = f \cdot \Delta\phi$, where $\Delta\phi = (\pi/2)/K$ is one band's own angular width and $f \in \{-0.4, -0.3, \ldots, 0.3, 0.4\}$ (nine values, $f = 0$ exactly reproducing the already-published ratio). Real cells are re-assigned to bands by the shifted boundary and the last-band/first-band area ratio $\rho^{\text{real}}_{K-1}/\tan(\pi/4K)$ is recomputed at each $f$. The perturbation scale is deliberately the band's own width, the coarsest scale at which a boundary shift can plausibly interact with *this* ratio's definition, not a sub-cell scale, which would simply redefine which physical band is being measured. Source: `code/ring_placement_boundary_aliasing_check.py`, `data/ring_placement_boundary_aliasing_results.json`.

## 66.2 Results

Two things are immediately visible and hold for all nine points without exception. First, the ratio is not a noisy or oscillating function of $f$: recomputing it at all nine tested fractions (not shown in full in the table, only the extremes) shows a monotonic, or very close to monotonic, decrease from $f = -0.4$ to $f = +0.4$ in every one of the nine configurations – rHEALPix's two coarsest configurations are the only cases showing brief flat plateaus, a consequence of very small $n_{K-1}$ (as few as 18 cells) making the band-membership function itself a coarse step function of $f$. Second, the magnitude of the swing is large and strikingly uniform across grids: every configuration moves from roughly 3.0–3.8 at $f = -0.4$ down to roughly 0.08–0.32 at $f = +0.4$, an order-of-magnitude range regardless of whether that grid's own baseline ($f = 0$) deviation was small (H3, $|\text{ratio} - 1|$ as low as 0.0033) or large (rHEALPix res 2, 0.2660).

Table 66.1: Real-over-theory ratio at the boundary shift extremes tested ($f = -0.4$ and $f = +0.4$, i.e. a shift of $\pm 40\%$ of one band's own angular width), against the already-published baseline at $f = 0$.

| grid | config | ratio ($f = -0.4$) | ratio ($f = 0$, baseline) | ratio ($f = +0.4$) |
|---|---|---|---|---|
| H3 | res 2, $K = 6$ | 3.269 | 0.968 | 0.268 |
| H3 | res 3, $K = 10$ | 3.255 | 1.003 | 0.264 |
| H3 | res 4, $K = 16$ | 3.248 | 1.009 | 0.261 |
| S2 | lvl 4, $K = 6$ | 3.430 | 0.936 | 0.323 |
| S2 | lvl 5, $K = 10$ | 3.205 | 0.831 | 0.284 |
| S2 | lvl 6, $K = 16$ | 3.208 | 0.957 | 0.267 |
| rHEALPix | res 2, $K = 6$ | 3.517 | 1.266 | 0.084 |
| rHEALPix | res 3, $K = 10$ | 3.843 | 0.840 | 0.235 |
| rHEALPix | res 4, $K = 16$ | 3.034 | 0.925 | 0.185 |

**Key Result**

Shifting the graticule band boundary by as little as 10%–40% of its own angular width changes the real/theory ratio by a full order of magnitude, uniformly across all three grids and all nine configurations tested, with a Pearson correlation of $r = 0.636$ between this sensitivity (measured as the ratio's range over the nine tested $f$ values) and each configuration's own baseline $|\text{ratio} - 1|$. The direction of the effect is not chaotic: it is smooth and monotonic in $f$, not the sign-flipping jaggedness a genuine per-cell aliasing signature would produce. This points to a geometric origin shared by all three grids – the last band is intrinsically thin ($\Delta\phi = (\pi/2)/K$, only $15°$ at $K = 6$), so a fixed *angular* shift removes or adds a proportionally large fraction of that band's own real estate no matter which grid's cells happen to fill it – rather than a grid-specific interaction between the boundary and each lattice's own discrete structure.

## 66.3 What this rules in, and what remains open

This chapter's finding sits between confirming and ruling out the aliasing candidate as originally framed in Chapter 64's todobox. The deviation *is* highly sensitive to exactly where the boundary is placed – so boundary placement is certainly part of the story, and the moderate positive correlation ($r = 0.636$) with baseline deviation is consistent with it contributing to items (iv) and (v). But the uniformity of the effect's magnitude across grids with very different cell geometries (H3's hexagons, S2's cubed-sphere quads, rHEALPix's diamonds and caps) argues that most of this sensitivity is a property of the graticule band construction itself near the pole – where every band is angularly thin by definition – rather than a signature of each grid's own discrete lattice interacting idiosyncratically with the boundary, which the original todobox item had proposed as the distinguishing mechanism.

**To Complete / Verify**

**Both items resolved: second in Chapter 69, first in Chapter 70.** This chapter used a single, coarse perturbation scale (fractions of the band's own width) and a uniform shift applied to *every* boundary simultaneously. Two narrower tests remain open. First, shifting only the single boundary between bands $K - 2$ and $K - 1$ (leaving all others fixed) would isolate whether the effect measured here is really about the last band specifically, or is an artefact of also perturbing band 0's lower edge at $\phi = 0$ in the same direction. Second, this chap-

ter's monotonic-not-chaotic finding somewhat undercuts the original "aliasing" framing (which predicts non-monotonic, resonance-like sensitivity as the boundary sweeps past individual cell edges one at a time); a finer sweep, at sub-cell rather than sub-band scale, restricted to the immediate neighbourhood of $f = 0$, has not been attempted and would be the more literal test of the aliasing hypothesis as originally stated, as opposed to the coarser boundary-placement-sensitivity test this chapter actually ran.

Chapter 67

# Mean Was Not Enough: The Band's IPQ Variance

## 67.1 Scope: the second item of Chapter 65's todobox

Chapter 65 correlated the polar band's *mean* IPQ with |ratio − 1| and found it tracks item (iv) (the cross-grid ordering: H3 closest, rHEALPix least close) but not item (v) (the resolution-to-resolution sign flip within a single grid), because mean IPQ is nearly flat across resolution within any one grid while the deviation is not monotonic. That chapter's own todobox proposed, as a distinct, narrower test, whether the band's IPQ *variance* – rather than its mean – tracks the resolution-to-resolution behaviour instead. This chapter runs that test.

**Methodological Note**

**What was done, precisely.** The per-cell IPQ lists already computed for Chapter 65 (`code/ipq_band_correlation_check.py`'s `h3_last_band_ipqs`, `s2_last_band_ipqs`, `rhealpix_last_band_ipqs`) are imported and reused unmodified – no new IPQ geometry is computed here. Only two things are new: the summary statistic taken over each list (population variance and standard deviation, via Python's `statistics` module, instead of the mean) and the correlation target, which is the *signed* deviation $\rho^{\text{real}}_{K-1}/\tan(\pi/4K) - 1$ rather than its absolute value, so that within-grid sign changes across the three resolutions can be examined directly against variance rather than only against its magnitude. Source: `code/ipq_band_variance_check.py`, `data/ipq_band_variance_results.json`.

## 67.2 Results

Table 67.1: IPQ variance and standard deviation of band $K-1$'s individual real cells, against the signed deviation $\rho^{\text{real}}_{K-1}/\tan(\pi/4K) - 1$ (recomputed independently here and matching Tables 63.1, 64.1 and 65.1 in magnitude).

| grid | config | $n_{K-1}$ | IPQ variance | IPQ stdev | signed dev. |
|---|---|---|---|---|---|
| H3 | res 2, $K = 6$ | 184 | $1.32 \times 10^{-6}$ | 0.00115 | −0.0319 |
| H3 | res 3, $K = 10$ | 478 | $7.46 \times 10^{-7}$ | 0.00086 | +0.0033 |
| H3 | res 4, $K = 16$ | 1,298 | $1.28 \times 10^{-7}$ | 0.00036 | +0.0086 |
| S2 | lvl 4, $K = 6$ | 48 | $7.67 \times 10^{-6}$ | 0.00277 | −0.0639 |
| S2 | lvl 5, $K = 10$ | 64 | $7.50 \times 10^{-7}$ | 0.00087 | −0.1694 |
| S2 | lvl 6, $K = 16$ | 120 | $2.12 \times 10^{-7}$ | 0.00046 | −0.0427 |
| rHEALPix | res 2, $K = 6$ | 18 | $9.66 \times 10^{-3}$ | 0.09829 | +0.2660 |
| rHEALPix | res 3, $K = 10$ | 50 | $4.56 \times 10^{-3}$ | 0.06752 | −0.1596 |
| rHEALPix | res 4, $K = 16$ | 162 | $3.31 \times 10^{-3}$ | 0.05751 | −0.0748 |

Cross-grid, IPQ variance correlates with |ratio − 1| at $r = 0.815$ across all nine points – stronger

than the mean-IPQ correlation of Chapter 65 ($r = -0.686$ in magnitude) – consistent with variance capturing something the mean does not: rHEALPix's band mixes near-circular authalic cap cells with far-from-circular skew quads, a within-band *shape* heterogeneity that a single mean discards but variance keeps.

Within each grid, across its own three resolutions, the pattern is mixed. H3's variance falls monotonically ($1.32 \times 10^{-6} \to 7.46 \times 10^{-7} \to 1.28 \times 10^{-7}$) while its signed deviation flips sign once ($-0.0319 \to +0.0033 \to +0.0086$); S2's variance also falls monotonically while its deviation stays negative throughout (no flip to explain); rHEALPix's variance falls monotonically while its deviation flips sign once, in the opposite direction from H3's flip ($+0.2660 \to -0.1596 \to -0.0748$). The within-grid Pearson correlations (variance vs. signed deviation, $n = 3$ per grid, reported as an exploratory pattern rather than a significance-tested statistic given the small $n$) are $r = -0.912$ for H3, $r = +0.297$ for S2, and $r = +0.930$ for rHEALPix – neither uniformly strong nor uniformly signed the same way across grids.

**Key Result**

IPQ variance is a better cross-grid predictor of deviation magnitude than IPQ mean was ($r = 0.815$ vs. $0.686$), extending Chapter 65's support for shape (over area) as the relevant property. But variance cannot, by construction, explain item (v)'s sign flips: variance is non-negative by definition and in every one of the three grids it decreases monotonically with resolution, while the signed deviation does not move monotonically in any of the three grids (flipping sign zero, one, or the reverse-one times respectively). A quantity that only ever decreases cannot be the mechanism behind a target quantity that changes direction – this is a structural, not merely empirical, limitation of variance as a candidate for item (v), disclosed here rather than obscured by reporting only the cross-grid correlation.

## 67.3 What this rules in, and what remains open

This chapter narrows item (iv) further (shape heterogeneity, not just shape mean, tracks the cross-grid ordering even better) while formally closing off IPQ variance as a candidate for item (v) specifically, for the structural reason given above rather than a weak correlation. The resolution-to-resolution sign flip of item (v) remains the volume's least-explained finding across every mechanism tested so far (area CV in Chapter 64, IPQ mean in Chapter 65, IPQ variance here, and graticule boundary placement in Chapter 66, which at least offers a mechanism – boundary sensitivity – capable in principle of flipping sign, unlike any per-cell summary statistic that is bounded on one side).

**To Complete / Verify**

Two directions remain open. First, a signed, rather than magnitude, shape statistic – for instance the band's IPQ *skewness*, which unlike variance is not sign-bounded and so could in principle track a sign-flipping target – has not been computed. Second, this chapter treated each grid's three resolutions independently; whether the *direction* of the resolution-to-resolution IPQ variance change (all three grids happen to decrease here, but this was not tested against a wider spread of resolutions per grid) is itself grid-invariant or coincidental to the particular three resolutions Table 63.1 originally tabulated remains unchecked.

Chapter 68

# A Signed Candidate: The Band's IPQ Skewness

## 68.1 Scope: the first item of Chapter 67's todobox

Chapter 67 tested the polar band's IPQ *variance* against item (v) (the within-grid, resolution-to-resolution sign flip of the signed deviation $\rho^{\text{real}}_{K-1}/\tan(\pi/4K) - 1$) and found it cannot, by construction, explain a sign-flipping target: variance is non-negative and fell monotonically with resolution in all three grids, while the signed deviation did not move monotonically in any of them. That chapter's own todobox proposed, as the natural next candidate, a *signed* shape statistic – the band's IPQ *skewness*, which unlike variance is not sign-bounded and so could in principle track a sign-flipping target. This chapter runs that test.

**Methodological Note**

**What was done, precisely.** The per-cell IPQ lists already computed for Chapter 65 (`code/ipq_band_correlation_check.py`'s `h3_last_band_ipqs`, `s2_last_band_ipqs`, `rhealpix_last_band_ipqs`) are imported and reused unmodified, exactly as Chapter 67 already did for variance – no new IPQ geometry is computed here. The only new computation is the summary statistic itself: Python's `statistics` module does not provide a skewness function, so the population (non-bias-corrected) Fisher–Pearson coefficient $g_1 = m_3/m_2^{1.5}$, with $m_k = \text{mean}((x - \bar{x})^k)$, is computed directly over the same per-cell lists. Source: `code/ipq_band_skewness_check.py`, `data/ipq_band_skewness_results.json`.

## 68.2 Results

Table 68.1: IPQ skewness of band $K-1$'s individual real cells, against the signed deviation $\rho^{\text{real}}_{K-1}/\tan(\pi/4K) - 1$ (recomputed independently here and matching Tables 63.1, 64.1, 65.1 and 67.1 in magnitude).

| grid | config | $n_{K-1}$ | IPQ skewness | signed dev. | signs match? |
|---|---|---|---|---|---|
| H3 | res 2, $K = 6$ | 184 | $-1.5660$ | $-0.0319$ | yes |
| H3 | res 3, $K = 10$ | 478 | $-2.0633$ | $+0.0033$ | no |
| H3 | res 4, $K = 16$ | 1,298 | $-0.6301$ | $+0.0086$ | no |
| S2 | lvl 4, $K = 6$ | 48 | $-0.4303$ | $-0.0639$ | yes |
| S2 | lvl 5, $K = 10$ | 64 | $-0.8821$ | $-0.1694$ | yes |
| S2 | lvl 6, $K = 16$ | 120 | $-1.1927$ | $-0.0427$ | yes |
| rHEALPix | res 2, $K = 6$ | 18 | $+2.4742$ | $+0.2660$ | yes |
| rHEALPix | res 3, $K = 10$ | 50 | $+3.1541$ | $-0.1596$ | no |
| rHEALPix | res 4, $K = 16$ | 162 | $+1.4993$ | $-0.0748$ | no |

Unlike variance, skewness is not sign-bounded, and its sign does vary across grids: negative

throughout for H3 and S2 at every one of the three tested resolutions, positive throughout for rHEALPix. But within each grid, across its own three resolutions, the skewness sign is *constant* – it never flips – while the signed deviation itself does flip within H3 (once) and within rHEALPix (once), and stays negative throughout only for S2. The result is that skewness's sign matches the deviation's sign pattern in exactly one of the three grids (S2, where the deviation itself never flips, so any grid-consistently signed quantity would match trivially there), for five sign-matches out of nine points overall.

**Key Result**

Five matches out of nine is not distinguishable from the four matches expected by chance alone under a uniform random sign assignment (binomial, $p = 0.5$, $n = 9$: $P(X \geq 5) \approx 0.50$). Skewness is not structurally excluded from explaining item (v) the way variance was – it is a signed quantity, and its per-grid sign is not fixed by construction the way variance's non-negativity is – but it fails *empirically*: within each grid it is one constant sign across all three resolutions tested, so it cannot itself be the mechanism producing a sign flip that happens *within* two of the three grids. The distinction from Chapter 67 is therefore one of failure mode, not of outcome: variance was ruled out structurally, skewness is ruled out empirically, but neither survives as a candidate for item (v).

## 68.3 What this rules in, and what remains open

This chapter closes Chapter 67's todobox's first item. Combined with Chapter 64 (area coefficient of variation), Chapter 65 (IPQ mean), and Chapter 67 (IPQ variance), four independent per-cell summary statistics of the polar band's shape and area have now been tested against item (v) and none explains it: two are structurally incapable (variance, and the coefficient of variation's own non-negativity) and two fail empirically despite being signed quantities in principle capable of it (IPQ mean, nearly flat within each grid, and now IPQ skewness, sign-constant within each grid). This strengthens, rather than weakens, the case made in Chapter 66 that item (v)'s resolution-to-resolution sign flip is more likely a graticule-boundary placement effect – a property of where the fixed band boundary falls relative to each grid's own lattice at each resolution, not a property of the cells' own area or shape distribution – since every per-cell statistic tested so far, signed or not, has failed to track it.

**To Complete / Verify**

This chapter treated each grid's three resolutions independently, as every chapter in this thread has; whether a genuinely different per-cell statistic (for instance a higher moment still, or a distributional-shape test rather than a single summary number) could succeed where mean, variance, and skewness have all failed remains untested and is not pursued further here, since the volume's own Chapter 66 already offers a mechanism-level candidate that these per-cell summary statistics were, by their nature, never going to be able to test directly.

Chapter 69

# Sub-Cell Resolution: Closing the Literal Aliasing Test

## 69.1 Scope: the second item of Chapter 66's todobox

Chapter 66 tested boundary-shift sensitivity at nine coarse fractions of the band's own width ($f \in \{-0.4, \ldots, +0.4\}$, step $0.1$) and found the real/theory ratio moves smoothly and monotonically with $f$, not with the non-monotonic, sign-flipping jaggedness a literal per-cell aliasing signature would produce. That chapter's closing todobox left open whether a much finer sweep, restricted to the immediate neighbourhood of $f = 0$ and resolved at sub-cell rather than sub-band scale, might still reveal such resonance at a scale the coarse nine-point sweep could not resolve – the more literal test of the aliasing hypothesis as originally framed. This chapter closes that item. (The chapter's other open item – isolating the single boundary between bands $K-2$ and $K-1$ from band 0's simultaneous shift – is a logically separate question and is not addressed here; see Section 69.3.)

**Methodological Note**

**What was done, precisely.** Two complementary arguments, both built directly on Chapter 66's own `band_index_shifted()` threshold logic and reusing unchanged its real per-cell geometry engines (H3, S2, rHEALPix latitude and area calls) for the same nine (grid, configuration) points.
(1) *A structural argument.* By `band_index_shifted()`'s own definition, a cell qualifies for the last band's area sum iff lat $\geq (K-1)\Delta\phi + \varepsilon$, and for band 0's sum iff lat $< \Delta\phi + \varepsilon$. As $\varepsilon$ increases, the last-band threshold rises, so the qualifying set (and the last-band sum) is non-increasing in $\varepsilon$; symmetrically the band-0 threshold also rises, so its qualifying set is non-decreasing in $\varepsilon$. The ratio of a non-increasing function to a non-decreasing one is itself non-increasing in $\varepsilon$ at *every* scale, by construction – no choice of sampling resolution, however fine, can expose a non-monotonic reversal, because none exists under this perturbation definition.
(2) *A cell-exact empirical confirmation.* Rather than sampling $\varepsilon$ on an arbitrary fine grid, which could straddle two step events and miss one, the exact threshold at which each individual edge cell enters or leaves its sum ($t_{\text{last}} = \text{lat} - (K-1)\Delta\phi$, $t_{\text{band0}} = \text{lat} - \Delta\phi$) is enumerated within a window of $\pm 2\Delta\phi$ around each edge, restricted to $|\varepsilon| \leq w$, where $w$ is set per configuration to $\max(4 \cdot \text{median gap between consecutive distinct crossing latitudes}, 0.01\Delta\phi)$ – a resolution-adaptive window guaranteeing a genuine double handful of real crossing events regardless of grid density. The ratio is evaluated exactly at the midpoint of every resulting plateau, and the number of downward vs. upward steps in the sequence is counted directly: the aliasing hypothesis predicts upward steps should appear; the structural argument in (1) predicts zero, at any resolution. Source: `code/ring_placement_subcell_sweep_check.py`,

`data/ring_placement_subcell_sweep_results.json`.

## 69.2 Results

Table 69.1: Cell-exact sub-cell sweep around $f = 0$ for the same nine (grid, configuration) points as Table 66.1. "events" is the number of real crossing thresholds found inside the window; "down"/"up"/"flat" partition the plateau-to-plateau ratio steps; $w/\Delta\phi$ is the window half-width as a fraction of one band's own angular width; the baseline column reproduces Table 66.1's $f = 0$ value exactly, as a consistency check.

| grid | config | $n_{\text{edge}}$ | events | down | up | flat | $w/\Delta\phi$ | baseline |
|---|---|---|---|---|---|---|---|---|
| H3 | res 2, $K = 6$ | 3,710 | 29 | 29 | 0 | 0 | 0.0100 | 0.9681 |
| H3 | res 3, $K = 10$ | 14,786 | 146 | 145 | 0 | 1 | 0.0100 | 1.0033 |
| H3 | res 4, $K = 16$ | 62,388 | 539 | 525 | 0 | 14 | 0.0100 | 1.0086 |
| S2 | lvl 4, $K = 6$ | 896 | 9 | 9 | 0 | 0 | 0.1689 | 0.9361 |
| S2 | lvl 5, $K = 10$ | 2,120 | 13 | 13 | 0 | 0 | 0.0591 | 0.8306 |
| S2 | lvl 6, $K = 16$ | 5,232 | 14 | 14 | 0 | 0 | 0.0196 | 0.9573 |
| rHEALPix | res 2, $K = 6$ | 302 | 11 | 9 | 0 | 1 | 2.2980 | 1.2660 |
| rHEALPix | res 3, $K = 10$ | 1,646 | 20 | 19 | 0 | 0 | 1.2642 | 0.8404 |
| rHEALPix | res 4, $K = 16$ | 8,174 | 21 | 21 | 0 | 0 | 0.6740 | 0.9252 |

Every one of the 802 real, cell-exact crossing events found across the nine configurations produces either a downward step (784 of them) or a flat step (16 of them, discussed below); not one produces an upward step. The baseline column reproduces Table 66.1's already-published $f = 0$ ratios exactly, confirming the two scripts' band-membership logic agrees at the one point they share.

**Key Result**

Zero upward steps occur in 802 real, cell-exact crossing events across all nine (grid, configuration) points, at a window scale as fine as 1% of one band's own angular width for the two densest grids (H3, S2). This is the literal, cell-exact confirmation of Section 69.1's structural argument: the ratio's monotonicity in $\varepsilon$ is not an artefact of the coarse nine-point sweep's sampling resolution, and it does not break down at finer scales. The non-monotonic, resonance-like signature a literal per-cell aliasing effect would produce does not exist under this perturbation definition, at any resolution – not because it was too fine to observe, but because it is structurally excluded by `band_index_shifted()`'s own monotone threshold construction.

Two details are flagged rather than smoothed over. First, the resolution-adaptive window $w$ is defined relative to each configuration's own median inter-cell crossing gap, not to $\Delta\phi$ directly; for rHEALPix at its two coarsest tested resolutions ($K = 6$, $K = 10$) the resulting window is *wider* than one full band ($w/\Delta\phi = 2.298$ and $1.264$ respectively), a consequence of those configurations having as few as 302 edge cells to draw crossing thresholds from. "Sub-cell" should be read as holding strictly for H3 and S2 at all three resolutions and for rHEALPix only at its finest ($K = 16$); at rHEALPix's two coarsest points the test is better described as sub-band rather than sub-cell, though it remains cell-exact and the zero-upward-step result holds regardless. Second, two of the 802 nominal steps (one each in

the rHEALPix $K = 6$ and $K = 10$ rows) are excluded from the down/up/flat tally because the corresponding plateau has zero cells in band $0$'s sum, making the ratio undefined at that midpoint rather than zero or negative; this is a boundary artefact of the same sparsity, not a counted upward step, and does not affect the headline count.

## 69.3 What this rules in, and what remains open

This chapter closes the second item of Chapter 66's todobox. Combined with that chapter's own coarse nine-point sweep, the ratio's monotonicity in the boundary shift $\varepsilon$ is now established at both the sub-band scale (fractions of $\Delta\phi$) and, for six of the nine configurations, at genuinely sub-cell scale down to 1%–6% of $\Delta\phi$ – with the structural argument in Section 69.1 guaranteeing the same holds at any finer scale still, including the three configurations where the empirical window could not itself be pushed below one full band width. The literal "aliasing" framing originally proposed in Chapter 64's todobox – resonance-like, non-monotonic sensitivity as the boundary sweeps past individual cell edges – is now ruled out, not merely undetected, as a description of this perturbation.

**To Complete / Verify**

**Resolved in Chapter 70.** This chapter addresses only the second of Chapter 66's todobox's two items. The first – shifting only the single boundary between bands $K - 2$ and $K - 1$, leaving band $0$'s edge fixed, to isolate whether that chapter's measured effect is really about the last band specifically or is partly an artefact of also perturbing band $0$ in the same direction – is not attempted here, since it tests a different perturbation (a single shifted boundary vs. this chapter's refinement of the uniform-shift sweep's resolution) and required its own script rather than a resolution refinement of this one.

Chapter 70

# Isolating the Last Boundary: Is Band 0 an Accessory?

## 70.1 Scope: the first item of Chapter 66's todobox

Chapter 66 shifted *every* band boundary $\phi_i = i\Delta\phi$ uniformly by the same $\varepsilon$, including band 0's own boundary at $\phi = \Delta\phi$, and found a large, order-of-magnitude swing in the real/theory ratio. Chapter 69 then confirmed that swing is smooth and monotonic in $\varepsilon$ down to sub-cell scale, ruling out literal per-cell aliasing as its mechanism. Neither chapter, however, addressed the remaining item of Chapter 66's own todobox: because the uniform shift moves band 0's boundary in the same direction at the same time, part of the measured swing could in principle be an artefact of also perturbing the denominator (band 0's area), rather than being entirely about the last band's own inner edge – the numerator. This chapter isolates the two by shifting *only* the boundary between bands $K-2$ and $K-1$, while band 0's definition (lat $< \Delta\phi$) is held exactly at its unperturbed position throughout.

**Methodological Note**

**What was done, precisely.** The same nine (grid, configuration) points and the same nine fractions $f \in \{-0.4, -0.3, \ldots, 0.3, 0.4\}$ of $\Delta\phi = (\pi/2)/K$ as Chapter 66's own coarse sweep, for direct point-by-point comparison. For each $f$, $\varepsilon = f\Delta\phi$, but now only the last band's inner edge moves: a cell qualifies for the last-band sum iff lat $\geq (K-1)\Delta\phi + \varepsilon$, while band 0's sum is computed once, at its fixed, unperturbed definition (lat $< \Delta\phi$), independent of $\varepsilon$ – so band 0's area is, by construction, identical to Chapter 66's own $f = 0$ baseline at every $f$ tested here. Real per-cell geometry (H3, S2, rHEALPix latitude and area calls) is reused unchanged from that chapter's own `h3_band_areas/s2_band_areas/rhealpix_band_areas`, a full population scan rather than an edge-windowed one, since $f$ ranges over $\pm 40\%$ of $\Delta\phi$, the same range that script already scans in full. Chapter 66's own published $f = 0, \pm 0.4$ ratios and full nine-point sweep (loaded directly from its own results JSON, not re-typed) supply the uniform-shift comparison column. Source: `code/ring_placement_last_boundary_only_check.py`, `data/ring_placement_last_boundary_only_results.json`.

## 70.2 Results

The $f = 0$ column reproduces Chapter 66's own published baseline exactly at all nine points, confirming the two scripts' band-0 accounting agrees at the one $\varepsilon$ where they must. Away from $f = 0$, every one of the nine last-boundary-only sweeps remains monotonic and roughly linear in $f$ – by the same structural argument as Chapter 69 (the last-band sum alone is non-increasing in $\varepsilon$, and here the denominator is a fixed constant, so the quotient inherits that monotonicity exactly). The range ratio in the final column is where the isolation actually bites:

Table 70.1: Real/theory ratio at the shift extremes when only the $K-2/K-1$ boundary moves (band $0$ fixed), against Chapter 66's own uniform-shift ratios at the identical nine $f$ values; "range" is max − min over all nine $f$ for each sweep, and the final column is the last-boundary-only range as a fraction of the uniform-shift range at that same point.

| grid | config | last boundary only | | | range | | range |
|---|---|---|---|---|---|---|---|
| | | $f=-0.4$ | $f=0$ | $f=+0.4$ | last-only | uniform | ratio |
| H3 | res 2, $K=6$ | 1.9934 | 0.9681 | 0.3737 | 1.6196 | 3.0006 | 0.540 |
| H3 | res 3, $K=10$ | 1.9531 | 1.0033 | 0.3678 | 1.5853 | 2.9912 | 0.530 |
| H3 | res 4, $K=16$ | 1.9518 | 1.0086 | 0.3655 | 1.5863 | 2.9863 | 0.531 |
| S2 | lvl 4, $K=6$ | 2.1383 | 0.9361 | 0.4481 | 1.6902 | 3.1066 | 0.544 |
| S2 | lvl 5, $K=10$ | 1.8485 | 0.8306 | 0.4009 | 1.4476 | 2.9203 | 0.496 |
| S2 | lvl 6, $K=16$ | 1.9826 | 0.9573 | 0.3681 | 1.6145 | 2.9407 | 0.549 |
| rHEALPix | res 2, $K=6$ | 3.5166 | 1.2660 | 0.1407 | 3.3759 | 3.4322 | 0.984 |
| rHEALPix | res 3, $K=10$ | 1.6471 | 0.8404 | 0.3025 | 1.3446 | 3.6079 | 0.373 |
| rHEALPix | res 4, $K=16$ | 1.9305 | 0.9252 | 0.2856 | 1.6449 | 2.8488 | 0.577 |

for seven of the nine configurations it clusters between $0.50$ and $0.58$, meaning roughly half of the uniform sweep's swing survives when band $0$ is held perfectly still. The remaining two are outliers in opposite directions – rHEALPix's coarsest configuration ($K=6$, $108$ cells in band $0$) retains $98.4\%$ of the uniform swing, while its next configuration ($K=10$, $756$ cells) retains only $37.3\%$.

**Key Result**

Isolating the shift to only the last band's own inner edge preserves, median across the nine configurations, $54.0\%$ of the uniform-shift sweep's range (mean $56.9\%$). The effect is therefore neither a pure artefact of also perturbing band $0$ – more than half the swing survives in seven of nine configurations even with band $0$ held exactly fixed – nor is band $0$'s simultaneous shift in Chapter 66 negligible: the uniform sweep's range is on the order of double the isolated one in most configurations, meaning band $0$'s own shift contributes roughly as much to the total swing as the last band's does. Chapter 66's original finding – that the deviation is highly sensitive to boundary placement – is thus confirmed to be genuinely, substantially about the last band's own edge, not merely an artefact of the denominator also moving, though the denominator's motion is not negligible either.

The two outliers are not smoothed into the median. Both occur at rHEALPix's two sparsest, coarsest configurations, where band $0$ holds as few as $108$ and $756$ cells respectively (an order of magnitude fewer than every other point in the table); with that few cells, a single cell's worth of area can dominate the band-$0$ sum, making the range-ratio at these two points a property of which particular cells happen to fall inside the fixed band-$0$ threshold at this resolution, rather than a stable signal comparable across grids. The $0.50$–$0.58$ clustering for the other seven configurations – spanning both dense grids (H3, S2, thousands to tens of thousands of band-$0$ cells) and rHEALPix's own finest tested resolution ($3{,}564$ band-$0$ cells) – is treated as the more reliable estimate of the isolated effect's true magnitude.

## 70.3 What this rules in, and what remains open

This chapter closes the first, and last remaining, item of Chapter 66's todobox. Together, Chapters 66, 69 and 70 now establish that the last-band/band-0 ratio's sensitivity to graticule boundary placement is: monotonic rather than resonant, at every scale from sub-band down to sub-cell (Chapter 69); genuinely, substantially attributable to the last band's own inner edge specifically, not merely to the denominator also moving (Chapter 70, this chapter); and yet not *purely* a last-band effect either, since band 0's own simultaneous shift contributes a comparable share of the total swing in the original uniform-shift framing. No further narrowing of the boundary-placement-sensitivity mechanism is proposed by either of this chapter's two predecessors, and none is opened here.

Chapter 71

# An Empirical Case Study: Hexagonal Grids and Generalization in Geospatial Deep Learning

## 71.1 From geometry to a downstream cost

Chapters 1 and 58 established that a lat/lon discrete graticule carries a genuine, exactly quantifiable non-uniformity between equatorial and polar cells, with the polar degeneracy decaying more slowly ($\Theta(K^{-1})$) than the typical bulk error ($\Theta(K^{-2})$). That is a purely geometric statement. This chapter asks whether the same non-uniformity has a measurable downstream cost in a concrete application, using a 2026 study that supplies exactly this kind of evidence in a domain *No Map Is Best* did not touch: geospatial deep learning.

## 71.2 The systematic sampling-bias problem

A 2026 study demonstrates a concrete case of the "extrapolation failure" pattern examined in Ch. 37 of Monograph 4, but in a different application domain: training deep learning models on Earth Observation data organized on a lat/lon grid suffers from a systematic sampling bias that is accentuated at high latitudes (Author, s). This is the machine-learning-facing counterpart of exactly the cell non-uniformity quantified in Proposition 1.2: a lat/lon grid oversamples high-latitude cells relative to their true physical extent by a factor that Proposition 58.1 shows diverges as $1/\rho_{\min}(K) \sim 4K/\pi$ near the pole.

The authors report that using a hexagonal discrete global grid (HDGGS, built on an ISEA projection) reduces model RMSE in high-latitude regions by approximately 4.8%–14.1% relative to a comparable model trained on a traditional lat/lon grid, with a corresponding reduction in the spread (range and standard deviation) of accuracy across different latitude bands (Author, s).

**Methodological Note**

The paper's full record — *Hexagonal discrete global grid system enhances model latitudinal spatial generalization and deep learning capabilities*, *Big Earth Data*, DOI 10.1080/20964471.2026.2666953, received 9 January 2026, accepted 5 April 2026, published online 20 May 2026 — has been directly verified against the publisher's abstract page. The authors' names, however, do not appear in any of the openly accessible abstract or metadata excerpts consulted so far; the entry in Section 114.6 lists the paper without an author field rather than guessing, and no citation beyond the abstract-level content above should be treated as verified until the full text or a complete metadata record is located.

**To Complete / Verify**

Confirm the authors' names and full editorial detail (volume, issue, page range) for Author (s). Until then, do not cite further numeric detail from this paper beyond the RMSE reduction range and the qualitative claim about band-wise variance already stated above, both of which come directly from the publicly accessible abstract.

## 71.3 Why this belongs in the global frame, not the regional case

It might seem that a hexagonal ISEA grid is itself a kind of *regional* fix (Part II of this volume treats genuinely regional, extent-limited optimization). It is not: the HDGGS of Author (s) still covers the whole globe, at every latitude; what changes relative to the lat/lon graticule of Chapter 1 is the choice of base tessellation, not the spatial extent over which it is optimized. It therefore belongs squarely in Part I's global register: it is a second global tessellation scheme, alongside the toy lat/lon graticule and the five implementations of Kmoch et al. (2022), evaluated against the same underlying question (how uniform are cell area and shape at global scale) but through a downstream machine-learning metric rather than a purely geometric one.

**To Complete / Verify**

Left open: a direct quantitative bridge between Proposition 58.1's exact geometric boundary rate and the 4.8%–14.1% RMSE reduction reported empirically here. The two quantities are not obviously on the same scale (one is a ratio of cell areas, the other a reduction in a model's prediction error), and no functional relationship between them should be asserted without an explicit, checked derivation or a reproduction of the original experiment.

Chapter 72

# Hexagon Mosaic Maps: From a Sampling Grid to a Display Convention

## 72.1 Scope: the same hexagon, read as a display surface rather than a sampling frame

Chapter 21 already cited White et al. (1992) once, in passing, for the truncated icosahedron's role in a pre-ISEA global sampling design. This chapter returns to that same 1992 paper and to the design it describes — the U.S. Environmental Protection Agency's Environmental Monitoring and Assessment Program (EMAP) hexagon grid — but asks a different question of it: not how the grid was built, but how data collected on it were *shown*. The answer takes this volume somewhere Chapter 71 did not go: from geometry and machine-learning generalization to thematic cartography proper, the discipline of encoding a statistical variable, and sometimes two variables at once, inside a tessellating cell so that a human reader can extract a pattern from it. The hexagonal tessellation reappears here for a third distinct reason within Part I — first as a DGGS base cell (Chapters 19 and 21), second as an input representation for deep learning (Chapter 71), and now as a display convention in its own right, predating both other uses by decades.

**Methodological Note**

This chapter introduces four new sources to Section 114.6 (Carr et al., 1992; Carr, 1990; U.S. EPA, 1992–2016; Kawakami et al., 2024) and performs no new computation of its own. Numeric claims below are taken either directly from a primary EPA source (the EMAP glossary's per-cell area figure) or from the abstract-level and citing-literature record of Carr et al. (1992) and Kawakami et al. (2024); neither paper's full text was fetched in this pass, and Kawakami et al. (2024) is an arXiv preprint whose peer-reviewed publication venue, if any, was not confirmed here. Both limitations are flagged again at point of use.

## 72.2 Two companion papers, one grid, one journal volume

White et al. (1992) and Carr et al. (1992) were published in the same year, in the same journal — *Cartography and Geographic Information Systems* volume 19, issues 1 and 4 respectively — and share an author (Denis White). This is not a coincidence of citation convenience: both papers describe the same underlying object, the EMAP hexagon grid, from opposite ends of a single pipeline. White et al. (1992) specifies how the hexagons are built and placed — the sampling-design half. Carr et al. (1992) specifies how a variable measured inside each of those hexagons, once the survey has been carried out, should be drawn on a map — the display half. A primary EPA source confirms the grid's own terms for what Chapters 19 and 21 would recognize as a DGGS in miniature: EMAP's internal glossary defines a tessellation as "a pavement or tiling of a space by a mosaic pattern" and states that "the EMAP design prescribes a regular tessellation

on the United States, consisting of hexagons of 634.5 km$^2$" (U.S. EPA, 1992–2016). The word "mosaic" in that EPA glossary entry is the same word Carr et al. (1992) put in their paper's title one year earlier; the display technique this chapter now turns to was named for the grid it was designed to visualize.

**To Complete / Verify**

The commonly cited figure of approximately 7,000 EMAP hexagons tessellating the world's land area, reported in secondary literature reviewing EMAP's twenty-first-century status, was not independently verified here against a primary EMAP source and is not used for any numeric claim in this chapter; only the directly sourced per-cell area for the U.S. design (634.5 km$^2$, U.S. EPA, 1992–2016) is treated as established.

## 72.3 The hexagon mosaic map and the ray glyph

Carr et al. (1992) use the phrase "hexagon mosaic map" for a map that tessellates a land mass with hexagonal cells and encodes a statistical variable, typically by color, inside each cell; the resulting map is described in the paper, and in every abstract-level record of it consulted for this chapter, as visually comparable to a color-contour (isarithm) map, in that both are built to reveal broad regional patterns rather than to let a reader read off an exact value for a specific point. Two further devices, both introduced in the same paper, extend the plain hexagon mosaic map beyond a single color-coded value per cell:

- The **ray glyph**: an oriented line segment with a dot at its base, placed inside a hexagon cell, whose *angle* encodes a local statistical estimate for that cell — a second, independent visual channel from the cell's fill color, occupying the same cell without requiring the cell to be subdivided.
- The **confidence-bounded ray glyph**: a direct extension in which upper and lower confidence bounds around the same estimate are drawn as a shaded arc, bounded by two rays, around the central ray — uncertainty made visible in the same glyph that carries the point estimate, rather than relegated to a separate map or a footnote.

Carr et al. (1992) extend the glyph once more for the bivariate case: the **bivariate ray glyph**, which the paper's own abstract describes as providing "a continuous representation for showing the local correlation of two variables" within a single cell — the paper's answer to a question this volume has not previously had to ask, of how a *relationship* between two quantities, not just either quantity alone, can be shown at the resolution of a single tessellating cell rather than only as a single scatterplot summarizing an entire study region. The worked example in Carr et al. (1992), consistent with the EMAP program's own early focus (Section 72.2), is a set of hexagon mosaic maps of acidic-deposition data for the eastern United States, the same class of environmental-monitoring variable EMAP's sampling design (White et al., 1992) was built to collect.

**To Complete / Verify**

The geometric construction of the ray glyph and the bivariate ray glyph — the precise mapping from a local estimate, or a local bivariate correlation, to a ray angle and an arc width — is described in Carr et al. (1992)'s full text, which was not fetched in this pass. No formula is reproduced here; only the qualitative description available consistently across the abstract and the multiple independent citing sources consulted for this chapter is used.

## 72.4 Why hexagons: the tessellation choice, argued twice

Carr et al. (1992) did not invent the idea of binning geographic data into a regular tessellation for display; Carr (1990), a 1990 Pacific Northwest National Laboratory report by the same first author on visualizing large data sets, had already explored binned data plots using both square and hexagonal cells two years earlier. What the 1992 paper adds, according to the secondary literature that discusses both papers together, is a direct, paper-length comparison of square against hexagonal tessellations specifically for the univariate and bivariate mosaic-map task (Kawakami et al., 2024). The recurring argument for hexagons over squares, attributed to Carr et al. (1992) and to the several later publications that cite them approvingly on this point, is perceptual rather than purely geometric: square tessellations introduce salient horizontal and vertical grid lines that compete for a viewer's attention with the data encoded in each cell, a distraction hexagonal tessellation avoids because no two hexagon edges in a standard orientation align with either map axis (Kawakami et al., 2024). This is a different argument from the ones Chapters 11 and 19 already gave for hexagons at DGGS scale — fewer distinct neighbor-distance classes, higher isoperimetric quotient, no orientation ambiguity of the kind Chapter 21 documents for square-derived schemes — and it is worth keeping the two arguments separate: Carr et al. (1992)'s case is about what a human visual system does when it looks at a gridded map, not about any property of the cells that survives independently of a viewer.

**Key Result**

The hexagonal tessellation is independently motivated three times in this volume, by three different disciplines, for three different reasons that happen to converge on the same shape: a sampling statistician's equal-area, low-edge-effect design criterion (White et al., 1992, Chapter 19), a cartographer's perceptual argument against competing grid-line orientation (Carr et al., 1992, this chapter), and a computational-geometry argument about neighbor-distance uniformity and compactness (Sahr et al., 2003; Kmoch et al., 2022, this volume's own reproduction chapters). None of the three arguments derives from, or depends on, either of the other two.

## 72.5 A 2024 descendant: hexagonal tiles, semantic icons, and an explicit MAUP link

Kawakami et al. (2024) present HexTiles, a hexagonal-tiling visual-encoding design for multivariate geospatial data built directly on Uber's H3 hierarchical hexagonal grid, the same H3 already documented at length in this volume's own reproduction chapters (Chapters 11, 12 and 25). The paper traces its own lineage explicitly back to Carr (1990) and Carr et al. (1992) for the choice of hexagons over squares (Section 72.4), then adds two design elements that were not available to the 1992 paper: *semantic icons*, small repeated symbols whose count within a cell encodes a variable in place of, or alongside, a color fill, chosen to exploit a viewer's ability to subitize small counts (recognize one, two, three, or four objects without counting); and an explicit *confidence encoding*, a spatially weighted variance computed for each aggregated value and rendered as an uncertainty-palette color desaturation, an inner-ring thickness, or an icon opacity, depending on which of the design's three visual channels carries that variable (Kawakami et al., 2024).

The confidence encoding is where Kawakami et al. (2024) make their connection to MAUP explicit, in terms this volume's own Chapter 24 already uses: the paper states its motivation

as addressing the fact that "MAUP can lead to the same aggregated result" from "entirely different" underlying data, and reports a worked example in which a hexagonal tile straddling a boundary between two very different underlying regions returns an identical spatially weighted mean in both cases, a mean the paper's own figure describes as "an incomplete picture of the data" without the added variance channel (Kawakami et al., 2024). That is precisely the aggregation half of the scale/zonation decomposition Chapter 24 traces to Openshaw (1984): a single number reported for a fixed cell can be consistent with radically different distributions of the underlying variable within that cell, and nothing about the cell's shape or fixed, reproducible boundary — this volume's repeated point about DGGS cells throughout Chapter 24 — prevents that. Kawakami et al. (2024)'s response is not to relocate the problem, in Chapter 24's sense, but to make the volume's own methodological discipline visible on the map itself: report the aggregated estimate *and* a directly computed measure of how much that estimate can be trusted at the resolution shown, which is exactly what Carr et al. (1992)'s confidence-bounded ray glyph, thirty-two years earlier, already did for a univariate estimate and its confidence bounds (Section 72.3).

**Methodological Note**

The evaluation results reported for HexTiles — a seven-participant user study and a three-person domain-expert review, comparing HexTiles against square-glyph maps on election and water-resource datasets — are taken from Kawakami et al. (2024)'s own reported statistics (a Mann–Whitney U test and a two-sample Z test for proportions, none significant at $\alpha = 0.05$, alongside a Levene's test difference in variance that was significant). These are the authors' own results, not independently reproduced or re-analyzed in this chapter, and are reported here only to describe what the paper claims to have found, not to endorse a particular effect size.

## 72.6 Synthesis: the same cell, asked to carry an estimate and its own uncertainty

Chapter 71 showed that a hexagonal DGGS carries a measurable downstream benefit for a machine-learning model trained on data organized on it, relative to a lat/lon graticule. This chapter shows a structurally similar but much older story, in a different application domain and for a different underlying reason. A hexagon tessellating a study area was first specified for reasons of sampling statistics (White et al., 1992), then given a companion display convention built around exactly that grid, one designed so that a single tessellating cell could carry an estimate, that estimate's uncertainty, and, in the bivariate extension, a local relationship between two variables, all without the grid lines themselves competing for a viewer's attention the way a square grid's aligned edges do (Carr et al., 1992). Thirty-two years later, and independently of the DGGS literature this volume has followed since Chapter 1, a hexagonal hierarchical grid built for an entirely different original purpose — Uber's H3, designed for ride-hailing logistics rather than environmental sampling — was adopted for exactly the same display task, and the same two-part solution reappeared: an aggregated value per cell, plus an explicit, separately computed confidence signal attached to it, this time named in the vocabulary of MAUP rather than of classical sampling statistics (Kawakami et al., 2024). Nothing in this chapter suggests the later authors were aware of the 1992 precedent's confidence-bounded ray glyph specifically; what the two papers share is a hexagon, a variable, and the same unavoidable cartographic problem — how to keep a summary statistic honest about what it does not know — solved twice, three decades apart, on the same tessellating shape this volume has spent Part I examining for entirely different, purely geometric reasons.

**To Complete / Verify**

A natural extension, not attempted here, would recompute Kawakami et al. (2024)'s weighted-variance confidence measure over an actual DGGS cell hierarchy already documented in this volume (H3 or rHEALPix, at the resolutions used in Chapter 11), to check whether the variance-based confidence signal behaves consistently across the exact cell populations and area distributions this volume has already tabulated. No such computation has been run, and no numeric outcome is asserted.

Chapter 73

# Before the Map: Hexagonal Binning as a Non-Geographic Precursor

## 73.1 Scope: completing a genealogy left incomplete in the previous chapter

Chapter 72 traced the hexagon mosaic map back to Carr et al. (1992) and, one step further, to Carr (1990)'s 1990 report on binned data plots. That report is not, however, where hexagonal binning itself began. Carr (1990) already had a three-year-old technique to draw on: Carr et al. (1987), a paper about scatterplot matrices that has nothing to do with geography at all. This chapter goes back one step further than Chapter 72 did, to a paper whose subject is overplotted statistical scatterplots, not maps, and asks why a technique built to solve a problem with no geographic content whatsoever ended up, five years later, naming a map.

**Methodological Note**

This chapter introduces one new source to Section 114.6 (Carr et al., 1987) and performs no new computation. The isoperimetric-quotient values invoked in Section 73.3 are not recomputed here; they are the same values already reported and sourced in Table 11.2 via Kmoch & Vasilyev (2022), reused for a different, purely two-dimensional argument than the one Chapter 11 originally made with them.

## 73.2 The 1987 problem: too many points, not too much geography

Carr et al. (1987) address a problem internal to exploratory statistics: when a scatterplot or a scatterplot matrix is drawn from a large number of observations $N$, individual points overlap so heavily that the plot no longer shows a distribution, only a saturated smear near the densest regions and an uninterpretable scatter of overplotted dots everywhere else. Their proposed remedy replaces individual points with a regular tessellation of the plotting plane, each cell shaded or sized according to a count or a density of the points falling inside it — density represented by gray-scale or by symbol area, in the paper's own framing, rather than by drawing every point. The tessellating shape the paper settles on, and the shape every later citing source credits it with introducing for this purpose, is the hexagon. Nothing in this construction refers to the Earth, a projection, or a geographic coordinate: the $x$- and $y$-axes of Carr et al. (1987)'s plots are two arbitrary statistical variables, and the technique would apply equally to any pair of numeric measurements with no spatial interpretation at all.

**To Complete / Verify**

The specific algorithmic details of Carr et al. (1987)'s binning procedure — exact cell-count formulas, the choice between gray-scale and symbol-area encoding in a given figure, and

the paper's own worked examples — are described in the paper's full text, which was not fetched in this pass; only the technique's existence, its hexagonal tessellation, and its motivating overplotting problem, as consistently reported across the citing literature consulted here (software documentation for R's `hexbin` package, matplotlib, and multiple independent expository sources), are used below.

## 73.3 Why hexagons for binning a plane: the same argument as Chapter 11, one dimension down

The reason given in the citing literature for choosing hexagons over squares in this purely planar, non-geographic setting is the same geometric property Section 11.3 of Chapter 11 already quantified for a sphere tessellated by DGGS cells: among the three regular polygons that tile a plane without gaps or overlaps — triangle, square, and hexagon, the only three for which this is possible — the hexagon has the highest isoperimetric quotient, meaning it is the closest of the three to a circle for a given area. Table 11.2 already gives the exact figure this volume needs for that comparison, established there for spherical DGGS cells but equally valid as a statement about plane geometry: a regular hexagon's isoperimetric quotient is $\approx 0.9069$, against $\pi/4 \approx 0.7854$ for a square, with $1$ representing a circle exactly. A hexagonal bin therefore aggregates points lying at more uniform distances from its own center than a square bin of the same area does, and — the property Chapter 72 already flagged as distinct from this one — a hexagon has six edge-adjacent neighbors at a single common distance, rather than a square's four edge-neighbors and four more distant corner-neighbors, so that a density estimate built by comparing a cell to its neighbors is not distorted by two different neighbor classes at two different distances.

**Key Result**

The argument for hexagons over squares that motivates Carr et al. (1987)'s scatterplot binning in 1987, and the argument that motivates H3, DGGRID's ISEA7H, and OpenEAGGR's ISEA3H as the three best-compacted schemes in Table 11.2, are the identical isoperimetric argument applied at two different dimensions of the same underlying problem: partitioning a two-dimensional surface — a flat statistical plane in one case, a sphere's local tangent neighborhood in the other — into cells that distort a point's true position within its cell as little as possible. Carr et al. (1987) reached this argument for an entirely statistical reason, five years before Carr et al. (1992) reached a cartographic version of the same argument, and roughly three decades before Chapter 11's own DGGS reproduction work reached a spherical version of it independently.

## 73.4 From a 1987 statistical technique to a name in software

The technique's later software history, unlike its 1987 origin, is tied together explicitly by shared authorship rather than by independent rediscovery. The R package `hexbin`, still distributed on CRAN, credits Daniel Carr as originating author, with Nicholas Lewin-Koh and Martin Mächler as later maintainers and porters (Carr et al., 2024); its own documentation cites Carr et al. (1987) directly as the method's source. The same technique, under the same name, now ships as `geom_hex` in the `ggplot2` plotting library — itself built on `hexbin` — as `plt.hexbin` in Python's Matplotlib, and as a general-purpose hexagon-binning plugin for the

D3 JavaScript library, none of which retain any connection to geography specifically; all three are general statistical-graphics tools whose worked examples in their own documentation are as likely to bin two arbitrary numeric columns as anything with a latitude or a longitude.

**To Complete / Verify**

The precise release history and current authorship credits for `hexbin`, `geom_hex`, and the Matplotlib and D3 hexagonal binning implementations were consulted only at the level of package documentation and citation records; no attempt is made here to verify version-by-version changes, and no numeric benchmark of these tools' performance is asserted.

## 73.5 Synthesis: the technique found geography twice, not once

Chapter 72 described a hexagon mosaic map as a technique built for a specific grid, the EMAP hexagon tessellation, by a paper that named the technique for the map it was designed to serve. This chapter's genealogy shows that description was one step too short: the technique existed first, in 1987, for a problem that had nothing to do with maps, grids, or the Earth at all, and only became geographic when Carr (1990) and then Carr et al. (1992) applied the same hexagon, chosen for the same isoperimetric reason, to data that happened to be collected over a country's surface rather than in a scatterplot's coordinate plane. The technique then found geography a second time, independently of its own authors' original 1990–1992 work and independently of the DGGS literature this volume has followed since Chapter 1: modern hexagon-binned choropleth-style maps built with `geom_hex` or a Python `hexbin` call over real latitude/longitude point data are, geometrically, the exact same statistical hexagon-binning operation Carr et al. (1987) built for a scatterplot's $x$–$y$ plane, applied to a plane whose two coordinates happen to be geographic rather than arbitrary. The same hexagon this volume has independently justified for a sphere (Chapter 11), for a display convention built around a fixed sampling grid (Chapter 72), and now for an entirely abstract statistical plane with no geographic content whatsoever, keeps being reached by the same isoperimetric argument, regardless of which of the three settings poses the question.

**To Complete / Verify**

Whether Carr et al. (1987)'s original 1987 paper itself draws any explicit connection to geographic tessellation, or whether the geographic application in Carr (1990); Carr et al. (1992) was an independent extension by the same author rather than one the 1987 paper anticipated, is not established here; the 1987 full text was not fetched, and this chapter's synthesis rests only on the technique's documented existence and its later, independently sourced geographic uses.

Chapter 74

# Six Roads to the Same Tessellation: Closing Part One

## 74.1 Scope: the debt this chapter closes

No chapter of Part I has yet stepped back to ask what its historical chapters, read together, say about why recursive polyhedral tessellation keeps being reinvented. Chapter 73, the part's most recent chapter, closes only its own hexbin thread; it explicitly does not attempt this wider synthesis. Monographs 1–4 of this series (*No Map Is Best*) closed with exactly this kind of cross-volume synthesis, stepping back from the individual information-theoretic, metric-embedding, and extreme-value results to ask what the three lines of attack had in common. This chapter does the same for Part One's six independently sourced pre-2000 (or, for PYXIS, pre-2010) precursor tessellations — QTM, HTM, GBT, White's diamonds, PYXIS, and the icosahedral grid developed for numerical weather prediction (Chapter 51) — using only material this volume has already established, chapter by chapter, and adding no new primary-source reading of its own.

**Methodological Note**

**What this chapter is and is not.** Every specific factual claim below — an author, a date, a motivating problem, an aperture, a base polyhedron — restates a claim already made and sourced in the chapter cited alongside it; nothing here is read from a new primary source. Where a chapter's own reading left a precursor's originating motivation less explicit than for the others (as turns out to be the case for two of the six, Section 74.4), this chapter reports that gap rather than filling it with a plausible-sounding motivation invented for the purpose of a tidier table.

## 74.2 Six precursors, six stated starting problems

Table 74.1 lays out the six schemes Part I has examined as historically or structurally independent precursors to the aperture-7 and aperture-4 DGGS families this volume's later chapters (H3, DGGRID, OpenEAGGR, rHEALPix) build on or resemble.

## 74.3 What converges, and what does not

Read side by side, Table 74.1's clearest pattern is negative: none of the four best-documented cases (the Sadourny grid, QTM, HTM, GBT) arose from an attempt to model the Earth's surface as such.[1] The Sadourny grid's motivation was a finite-difference scheme's own conser-

[1] White's diamonds are a fifth case for which Chapter 55 has since established a stated motivation, but they are deliberately not folded into this "four" count. Unlike the other four, White's own stated purpose (Section 55.2 of

Table 74.1: The six pre-2000/2010 precursor tessellations examined in Part I, with the originating problem each chapter's own reading attributes to its source, and the base unit and aperture each uses.

| Scheme & chapter | Base unit / aperture | Year | Stated originating problem |
|---|---|---|---|
| Sadourny icosahedral grid (Chapter 51) | Icosahedron triangle mesh (vertices addressed), aperture 4 | 1968 | Numerical weather prediction: a finite-difference scheme on the sphere whose advection of vorticity exactly conserves total vorticity, total squared vorticity, and total kinetic energy; the icosahedral choice is also credited by later secondary sources with avoiding the pole problem of regular latitude–longitude grids. |
| QTM (Chapter 37) | Octahedron triangle, aperture 4 | 1988–89 | Locational data quality: no coordinate system made a location's own measurement precision explicit; recursive triangles chosen for computational economy of point-to-facet lookup ("trilocation"), needing no square roots. |
| HTM (Chapter 38) | Octahedron triangle, aperture 4 | 2001 | Indexing hundreds of millions of astronomical catalogue objects (SDSS) so relational-database B-tree machinery, not a bespoke spatial index, could answer sky-region queries; motivated by a database engineering problem, not by coordinate precision. |
| GBT (Chapter 45) | Hexagon, aperture 7 | 1982 | Pattern recognition and automatic target recognition in computer vision; hexagonal pixel arrays chosen for their circle-approximation and neighbour-access properties in image processing, not for terrestrial modelling, which came only later via Sahr's DGGS extensions. |
| White's diamonds (Chapters 48 and 55) | Diamond (paired triangles), aperture 4 | 2000 | Stated purpose (survey sampling, dynamic modelling, data storage/retrieval) plus a combinatorial design rationale: no single existing tessellation unit combined nesting, complete polyhedral tessellation, and radial symmetry at once; diamonds were proposed to combine all three. Found only on a second, targeted reading of White (2000)'s Introduction and Conclusions (Chapter 55), not on Chapter 48's first reading. |
| PYXIS (Chapter 49) | Hexagon, aperture 3 | 2004/2011 (patent) | Commercial multiresolution digital-earth spatial data ordering (the patent's own title: "close-packed, uniformly adjacent, multiresolutional overlapping spatial data ordering"); read from the patent rather than the inaccessible journal paper, so the underlying research motivation behind the 2009 algebraic treatment is not independently confirmed here |

vation properties under long numerical integration, with the icosahedral choice also credited, by later secondary sources, with sidestepping the pole problem; QTM's motivation was measurement-precision bookkeeping for point locations already given in some coordinate system; HTM's was database query performance for a fixed astronomical catalogue; GBT's was image-array addressing for pattern recognition, with no terrestrial referent at all until Sahr's later DGGS work, already examined in Chapters 46 and 47 adopted it. Chapter 37's closing discussion already noted this same point for the QTM/HTM pair alone, calling it two unrelated problems arriving twelve years apart at structurally convergent solutions; Table 74.1 shows the pattern is not limited to that pair, and is at its widest for the Sadourny grid, which predates every other precursor examined here by at least fourteen years. Four independent research communities — numerical weather prediction, cartographic data-quality engineering, astronomical database design, and computer-vision pattern recognition — each reached for a recursively subdivided polyhedral or planar tessellation as their solution, without, so far as either scheme's own construction shows, one inventor building on another's design work.

This last clause needs a qualification that a later chapter of this volume forces on the present one. Sahr et al. (2003), the founding DGGS survey read in full for Chapter 54, does not merely resemble the Sadourny grid, QTM, and White's diamonds in retrospect: its own Table 1 and reference list directly cite and classify Sadourny et al. (1968) and Williamson (1968) alongside QTM and White's diamond scheme, using a single Class I/Class II subdivision taxonomy drawn from Kenner (1976). So by 2003 the field that would become DGGS had already placed the two NWP grids and QTM/White's diamonds inside one classificatory family, rather than treating them as four unrelated inventions merely observed, after the fact, to look alike. Chapter 54 reports this as resolving, in the direction of documented connection, the lineage question left open in Chapter 51.

What Sahr et al. (2003) does *not* show is that Dutton or White read Sadourny's or Williamson's 1968 papers while designing QTM in 1988 or the diamond scheme in 2000; the paper trail runs forward from a 2003 survey applying a shared taxonomy backward across independently built schemes, not from the 1968 NWP papers forward into QTM's or White's own design process. The independence this section claims is accordingly independence *of invention* — no evidence in this volume shows QTM's construction drawing on the NWP grids — and not independence of subsequent classification: by the time the DGGS field had a name and a founding survey, it already read the NWP grids and these two later precursors as members of one family. This chapter's framing of "four communities converging without contact" should be read with that distinction in mind for the Sadourny/QTM/White cluster specifically; it is not disturbed for HTM or GBT, for which this volume has located no comparable pre-DGGS taxonomic linkage to the other precursors.

For White's diamonds specifically, the independence-of-invention qualification just stated needs narrowing further still. Chapter 55, revisiting White (2000) in full for its own stated purpose, found that the paper's Applications section cites the NWP-descendant atmospheric-modelling literature (Thuburn, 1997; Heikes and Randall, 1995a) and Sahr's own NWP-symmetry-aware work (Sahr and White, 1998) to justify building the diamond scheme compatibly with the equatorially symmetric ISEA grid. Unlike the Sahr et al. (2003) taxonomic link, which is a later survey's classification applied backward, this is a direct citation made in White's own 2000 text, at the time of original publication, into the descendant NWP tradition. So for White's diamonds — and, on the evidence assembled in this volume, only for White's diamonds among the four best-documented precursors — even independence of in-

Chapter 55) is explicitly about building global grids for earth-surface applications — survey sampling, dynamic modelling, environmental data storage — which is closer to an attempt to model the Earth's surface than to the other four's unrelated originating problems. This does not weaken the pattern this paragraph reports; it identifies White's diamonds as an exception with a different profile, not as a fifth confirming instance.

vention no longer holds without qualification: the diamond scheme's own published design rationale draws, in part, on NWP-tradition literature. It remains true, on the same evidence, that this citation trail runs through 1990s descendant papers rather than the 1968 founding papers themselves (Sadourny et al., 1968; Williamson, 1968), and that whether those 1990s papers in turn cite the 1968 originals has not been checked (Chapter 55's own todobox).

What does *not* converge is the choice of aperture and base unit. The Sadourny grid, QTM, HTM, and White's diamonds all use aperture-4 subdivision of triangular (or, for Sadourny, vertex-addressed triangular) faces; GBT and PYXIS both use hexagonal cells but at different apertures (7 and 3 respectively) and, per Chapter 49, different addition-table structures entirely (PYXIS's two alternating Class I/Class II matrices against GBT's single table). Chapter 46 and Chapter 50 together give the hexagonal, aperture-7 branch a degree of independent algebraic and geometric confirmation — an exact planar isoperimetric ceiling (Eq. (46.1)) and a proven ring isomorphism to the 7-adic integers — that this volume has not established for the aperture-4 or aperture-3 branches; Section 74.4 reports this asymmetry as a gap in this volume's own coverage, not as a claim that the aperture-4 or aperture-3 schemes lack comparable structure.

**Key Result**

Of the four precursor schemes for which Part I was able to establish an explicit originating motivation on its first pass (the Sadourny grid, QTM, HTM, GBT), none was born as an attempt to model the Earth: one was solving a numerical-integration conservation problem in atmospheric modelling, two were solving unrelated engineering problems (coordinate-precision bookkeeping; astronomical database indexing), and one was solving a pattern-recognition problem in computer vision. White's diamond scheme is a fifth case with a documented motivation, established only on the second, targeted reading Chapter 55 performed; unlike the other four, its own stated purpose was building earth-surface grids for environmental applications, so it is reported separately rather than folded into the "none aimed at modelling the Earth" pattern above. All five, together with PYXIS, nonetheless converge on the same general solution shape — recursive subdivision of a base polyhedron or planar unit into a hierarchy of cells with an algebraic addressing scheme — while diverging sharply on aperture (3, 4, or 7) and base unit (triangle, diamond, or hexagon). This convergence-on-structure/ divergence-on-parameters pattern is the closest analogue, within Part I, to the cross-cutting synthesis Monographs 1–4 built around the continuous-mapping problem. For the Sadourny/QTM/White cluster specifically, Section 74.3 qualifies "independent" as independence of invention: a 2003 survey (Sahr et al., 2003, read in full for Chapter 54) already classified the NWP grids alongside QTM and White's diamonds in one taxonomy, so the four schemes did not remain permanently unrecognized as related once the DGGS field named itself; they were merely built without one inventor drawing on another's work.

## 74.4 Two gaps this synthesis does not paper over

One of the six originating-motivation cells in Table 74.1 is weaker than the other five, and this chapter reports that honestly rather than inventing a matching motivation for symmetry. Chapter 49 disclosed, in its own opening, that the algebraic source most directly attributable to a research motivation, Vince & Zheng (2009), could not be fetched in full, so what this synthesis reports for PYXIS is the patent's commercial framing, not necessarily the full research motivation behind the underlying mathematics. The comparable gap this chapter originally reported for White's diamond scheme — that Chapter 48's reading of White (2000) extracted

the construction and composition rule in detail but not an explicit statement of the problem the diamond subdivision was built to solve — has since been closed by Chapter 55, which re-read the same paper's Introduction and Conclusions specifically for that question and found both a stated purpose and a more specific combinatorial design rationale (Table 74.1's White row is accordingly updated by that chapter's findings, not retroactively rewritten here).

**To Complete / Verify**

Two items remain open by this synthesis, one from the original version of this chapter and one added by Chapter 55's own reading. **(i)** A comparably explicit originating-motivation statement for Vince & Zheng (2009)'s PYXIS algebra has not been located; its full text remains unfetched for the reasons given in Chapter 49. **(ii)** Chapter 55 found that White's diamond paper cites Thuburn (1997) and Heikes and Randall (1995a) as part of its own Applications section, but left unchecked whether those 1990s papers themselves cite Sadourny et al. (1968) or Williamson (1968) directly; reading them in full would settle whether the citation trail this volume has traced runs all the way back to the 1968 founding papers or stops at their 1990s descendants. **(iii)** This chapter's convergence/divergence observation in Section 74.3 is qualitative, built by reading five chapters side by side; it is not a claim quantified by any statistic this volume computes, and a more rigorous treatment might attempt to formalize "convergence on structure" in a way that could itself be measured, which is left for future work.

# Part II

# The Regional Case: Low-Distortion Auxiliary Spheres

Chapter 75

# Kerkovits's Approach: Regional Rather Than Global Optimization

## 75.1 The rectifying auxiliary sphere

Part I treated distortion as a global property: a single tessellation (lat/lon graticule, ISEA hexagons, or any of the five DGGS of Kmoch et al. (2022)) evaluated over the whole sphere at once. This chapter turns to the opposite strategy: optimize an auxiliary surface for a bounded latitude band only, accepting that the result is unusable outside that band.

Kerkovits (2026) proposes a new low-distortion auxiliary sphere, to be used as the intermediate surface in an aphylactic (neither equal-area nor conformal) double mapping for regional maps. The key property the author reports is the near-total absence of linear distortion (average linear scale between 0.99999 and 1.00001) for a latitude band roughly 20° wide (Kerkovits, 2026).

**Methodological Note**

The bibliographic record for Kerkovits (2026) has now been fully verified, resolving the earlier open item in the draft: KN — Journal of Cartography and Geographic Information, DOI 10.1007/s42489-026-00216-5, received 9 January 2026, accepted 28 February 2026, published under a CC BY 4.0 licence, appearing in issue 76/2 (2026) alongside articles by Edler, Schiewe et al., Gründemann & Burghardt, Kühne, Budke & Morawski, and Keil et al., confirmed independently against both the Springer DOI landing page and the Swiss Society of Cartography's own listing of that issue's table of contents. The numerical values above (0.99999–1.00001; 20° band) are still taken from the abstract; they should be checked against the full text before being used in any further derived computation, but the editorial record itself is no longer an open item.

## 75.2 Historical context: rectifying and parametric spheres

Kerkovits situates the new sphere within a lineage of auxiliary spheres used to reduce a two-step ellipsoid-to-plane mapping to a sphere-to-plane one. Kerkovits (2026) recalls that Oscar S. Adams's 1921 study gave a comprehensive theory of auxiliary spheres, cataloguing earlier constructions and introducing a sphere with equidistant meridians (the "rectifying sphere" proper) as well as a parametric sphere obtained from the ellipsoid by an affine transformation that preserves distances along every parallel. A closely related construction, the conformal sphere, remains in active use today in the official map projections of several European countries, including the Netherlands, Switzerland, Czechia, and Hungary (Kerkovits, 2026). The conformal sphere's known weakness is that it is unusable for a whole-Earth map, since renumbering both latitude and longitude on the sphere produces overlapping regions near the 180° meridian; its distortion behaviour, however, is favourable precisely for regional maps, which

is the property Kerkovits (2026)'s new sphere pushes further.

**Methodological Note**

Kerkovits (2026)'s full derivation of the new sphere (as opposed to the historical auxiliary spheres recalled above) is now independently reproduced, equation by equation, in Chapter 76, resolving this item.

## 75.3 What "regional" costs, structurally

The reason this belongs in a volume about the global/discrete frame of Part I is structural, not incidental: Kerkovits (2026)'s sphere achieves its near-zero linear distortion *by construction* restricted to a band of finite angular width, in direct analogy to how Proposition 58.1 showed the discrete graticule's cell-area ratio degrades specifically as the ring approaches the pole. Both results say, in their respective registers, that a single surface (continuous auxiliary sphere, or discrete tessellation) cannot be uniformly good across the whole range of latitude; the only choice is *where* to spend the available distortion budget. Part I spent it uniformly (accepting worse behaviour at the poles in exchange for global coverage); Kerkovits (2026)'s approach spends it entirely within a chosen band, at the cost of having no defined behaviour outside it. Chapter 78 makes this trade-off explicit and quantitative.

Chapter 76

# Reproducing the Rectifying Sphere: An Independent Derivation

## 76.1 Scope: closing the open item of Chapter 75

**Methodological Note**

Chapter 75 deferred Kerkovits (2026)'s full derivation pending access to the article's complete text, checking only the abstract-level numerical claims. That access is now available: the article is open access (CC BY 4.0), and its full derivation and worked example have been retrieved from the publisher's own record. This chapter does not merely transcribe that derivation; it re-implements every equation from scratch in an independent script (`code/kerkovits_sphere_reproduction.py`) and checks the resulting constants against the paper's own worked example to the precision the paper itself reports, following the same derive-then-verify discipline used throughout this volume (Chapters 1, 58 and 59).

## 76.2 Setting: the ellipsoid-to-sphere step of a double mapping

Write $\Phi, \Lambda$ for geodetic latitude and longitude on the reference ellipsoid, and $\varphi, \lambda$ for their image on the auxiliary sphere. Let $a, b$ be the ellipsoid's semi-major and semi-minor axes, $e$ and $e'$ its first and second eccentricities, and $c$ its polar radius of curvature:

$$e = \sqrt{\frac{a^2 - b^2}{a^2}}, \qquad e' = \sqrt{\frac{a^2 - b^2}{b^2}} = \frac{e}{\sqrt{1 - e^2}}, \qquad c = \frac{a}{\sqrt{1 - e^2}}. \tag{76.1}$$

Following Kerkovits (2026, eq. 4), define

$$v(\Phi) = \sqrt{1 + (e')^2 \cos^2 \Phi}, \tag{76.2}$$

so that the prime-vertical and meridional radii of curvature are $N(\Phi) = c/v(\Phi)$ and $M(\Phi) = c/v^3(\Phi)$, and the signed meridian-arc length from the Equator is $\sigma(\Phi) = \int_0^{\Phi} M(\Phi')\,\mathrm{d}\Phi'$.

Restricting to spheres that send meridians to meridians and parallels to parallels, with longitude scaled linearly, $\lambda = n(\Lambda - \Lambda_0)$, the meridional and parallel linear scales $h, k$ (sphere length over ellipsoid length, radius $R$) are

$$h = \frac{R}{M(\Phi)} \frac{\mathrm{d}\varphi}{\mathrm{d}\Phi}, \qquad k = \frac{Rn \cos \varphi}{N(\Phi) \cos \Phi}. \tag{76.3}$$

The *rectifying* sphere is the member of this family with $h \equiv 1$ everywhere: solving $\mathrm{d}\varphi/\mathrm{d}\Phi = M(\Phi)/R$ gives $\varphi = \sigma(\Phi)/R + \kappa$ for a constant of integration $\kappa$. Adams's 1921 rectifying sphere is the case $n = 1, \kappa = 0$, $R$ fixed by matching the full meridian length; Kerkovits (2026) instead leaves $n$, $\kappa$, and $R$ free and chooses them to minimize distortion in the neighbourhood of a freely chosen standard parallel $\Phi_0$, exactly as Gauß (1843) had already done for the conformal sphere and Kerkovits (2025) for an authalic one.

## 76.3 Solving for the free parameters

With $h \equiv 1$ already forced, only $k$ needs to be brought to a second-order tangency with 1 at $\Phi_0$: $k(\Phi_0) = 1$, $k'(\Phi_0) = 0$, $k''(\Phi_0) = 0$. Substituting Eq. (76.3) into these three conditions and solving in sequence (full algebra in Kerkovits (2026, Sec. 4); reproduced independently, step by step, in `code/kerkovits_sphere_reproduction.py`) yields a closed-form chain:

$$\tan\varphi_0 = \frac{\tan\Phi_0}{v(\Phi_0)}, \tag{76.4}$$

$$n = \frac{\sin\Phi_0}{\sin\varphi_0}, \tag{76.5}$$

$$R = \frac{c}{v^2(\Phi_0)} = \sqrt{M(\Phi_0)\,N(\Phi_0)}, \tag{76.6}$$

$$\kappa = \varphi_0 - \frac{\sigma(\Phi_0)}{R}. \tag{76.7}$$

Equation (76.6) identifies $R$ as the radius of the ellipsoid's own osculating sphere at $\Phi_0$ – a check available independently of Kerkovits (2026)'s own derivation, since $\sqrt{MN}$ is a standard closed-form quantity. Equations (76.5) and (76.6) do not depend on which of the three tangency conditions ($h \equiv 1$, $h = 1/k$, or $h = k$) is imposed away from $\Phi_0$; only Eq. (76.7) is specific to the rectifying (as opposed to authalic or conformal) choice, which is exactly the structural remark Kerkovits (2026) draws attention to in their own closing discussion.

**Key Result**

The rectifying, authalic, and conformal low-distortion spheres share *the same* $n$, $R$, and spherical standard-parallel image $\varphi_0$ for a given $\Phi_0$ and ellipsoid; the three constructions differ only in the single constant $\kappa$, which fixes where the zero of arc length is placed along the meridian. This is not a coincidence of algebra internal to one paper – it follows directly from Eqs. (76.4) to (76.6) being derivable from the $k$-tangency conditions alone, before $h$ is ever specialized.

## 76.4 Independent numerical verification

**Methodological Note**

**What was computed.** Equations (76.1), (76.2) and (76.4) to (76.7) were implemented from scratch on the WGS84 ellipsoid ($a = 6{,}378{,}137$ m, $f = 1/298.257223563$), with $\sigma(\Phi)$ evaluated by numerical quadrature (`scipy.integrate.quad`) rather than any closed-form meridian-arc series, so that no algebraic step is shared with Kerkovits (2026)'s own working other than the defining equations themselves. The standard parallel $\Phi_0 = 45°$ matches the paper's own worked example, chosen there, and here, only because it is the value the paper reports numbers for.

The match is exact to all reported digits ($\kappa$ differing only in the eleventh decimal, attributable to quadrature versus closed-form integration of $\sigma$). Section 114.3's earlier disclaimer – that the numerical values in Chapter 75 were still abstract-level – is now resolved for the derivation itself; only the paper's worked-out radius-versus-authalic/conformal comparison figures of its Section 5 remain outside this chapter's scope, since those depend on Kerkovits (2025)'s own authalic-sphere derivation, not reproduced here.

Table 76.1: Independent reproduction of Kerkovits (2026)'s worked example ($\Phi_0 = 45°$, WGS84), against the paper's own reported values. Source: `code/kerkovits_sphere_reproduction.py`, `data/kerkovits_sphere_reproduction_results.json`.

| quantity | this volume's independent computation | Kerkovits (2026) |
|---|---|---|
| $R$ | 6,378,101.030 m | 6,378,101.030 m |
| $\varphi_0$ | $44° 57' 6.5271''$ | $44° 57' 6.5271''$ |
| $n$ | 1.000842082541 | 1.000842082541 |
| $\kappa$ | 0.002985268413 rad | 0.002985268412 rad |

### 76.4.1 The acceptance band

Kerkovits (2026) report that, using the Bayeva–Airy–Kavrayskiy local distortion measure $\varepsilon^2 = \ln^2 h + \ln^2 k$ (here $h \equiv 1$, so $\varepsilon^2 = \ln^2 k$) and the strict threshold $\varepsilon^2 < 2 \times 10^{-10}$ (corresponding to an average linear distortion under 1:100,000), the rectifying sphere at $\Phi_0 = 45°$ is acceptable between $34° 5' 56''$ and $55° 13' 15''$ – an interval a little under $21°$ wide, consistent with the abstract's "approximately $20°$" figure already used in Chapters 75 and 78. Solving $\varepsilon^2(\Phi) = 2 \times 10^{-10}$ independently (root-finding on $k(\Phi)$ computed from Eq. (76.3) with the constants of Table 76.1, not from any tabulated value in the paper) gives $34° 5' 55.76''$ to $55° 13' 14.75''$ – agreeing with the paper to within $0.25''$ at both ends, well inside plausible rounding of the reported figures.

**Key Result**

The $[34° 5' 56'', 55° 13' 15'']$ acceptance band used without derivation in Chapters 75 and 78 is now independently reproduced from the paper's own defining equations, not merely copied from its abstract. This band is used again, unchanged, in Chapter 80.

### 76.4.2 The pole-adapted variant and $n$'s range

Kerkovits (2026)'s Section 6 records several boundary figures for the $\Phi_0 = 45°$ case: the ellipsoidal latitude whose image sits at the sphere's own pole ($89° 58' 46''$), the image of the ellipsoidal South Pole under this Northern-standard-parallel parametrization ($-89° 40' 43''$), and the general bound $1 \le n \le \sqrt{1 + (e')^2}$ across all choices of $\Phi_0$. All three were recomputed independently here (inverting Eq. (76.7) by root-finding for the first two; direct evaluation of $\sqrt{1 + (e')^2}$ on WGS84 for the third) and agree with the paper to within $0.4''$ and 6 significant figures respectively – see `data/kerkovits_sphere_reproduction_results.json` for the full figures. These confirm, independently, Kerkovits (2026)'s own recommendation that any regional band including a pole should be built with $\Phi_0 = \pm 90°$ rather than the general-latitude parametrization used elsewhere in this chapter.

### 76.4.3 Against the traditional spheres

**Methodological Note**

**What this is, and what it is not.** Kerkovits (2026)'s own Figs. 1–2 compare their optimized sphere against the traditional rectifying, authalic, and conformal spheres; those figures are not transcribed here, since they were never retrieved as data. What follows instead is an independent derivation and computation of the three traditional spheres from their own classical defining equations — Adams's 1921 rectifying sphere of Section 75.2, and the authalic and conformal constructions of Chapter 28 — each evaluated at the same five points across the $\Phi_0 = 45°$ band already fixed in Section 76.4.1, against the already-verified optimized sphere of Table 76.1. None of the three traditional spheres is uniquely forced by classical theory alone; the convention adopted for each is disclosed in `code/kerkovits_vs_traditional_spheres.py`: the rectifying sphere's radius matches the total meridian length exactly ($R = 6{,}367{,}449.146$ m); the authalic sphere uses the WGS84 authalic radius $R_A = 6{,}371{,}007.181$ m of Chapter 27; and the conformal sphere uses $R = a = 6{,}378{,}137.000$ m, the common "spherical Mercator" choice, since the classical theory fixes the conformal *latitude* $\chi$ but not a unique sphere radius. All three share $n = 1$ and $\kappa$ fixed by $\varphi(0) = 0$, and the authalic and conformal $h$ values are obtained by central-difference differentiation, matching the numerical-Jacobian discipline already used in Chapter 31.

Table 76.2: Local distortion $\varepsilon^2 = \ln^2 h + \ln^2 k$ of Kerkovits (2026)'s optimized sphere against the three traditional auxiliary spheres, across the $\Phi_0 = 45°$ acceptance band. Source: `code/kerkovits_vs_traditional_spheres.py`, `data/kerkovits_vs_traditional_spheres_results.json`.

| $\Phi$ | optimized (Table 76.1) | traditional rectifying | traditional authalic | traditional conformal |
|---|---|---|---|---|
| 34.0988° | $2.000 \times 10^{-10}$ | $1.327 \times 10^{-6}$ | $1.180 \times 10^{-6}$ | $2.195 \times 10^{-6}$ |
| 40° | $2.025 \times 10^{-12}$ | $9.733 \times 10^{-7}$ | $8.653 \times 10^{-7}$ | $3.799 \times 10^{-6}$ |
| 45° | $4.930 \times 10^{-32}$ | $7.073 \times 10^{-7}$ | $6.287 \times 10^{-7}$ | $5.570 \times 10^{-6}$ |
| 50° | $2.423 \times 10^{-12}$ | $4.835 \times 10^{-7}$ | $4.297 \times 10^{-7}$ | $7.684 \times 10^{-6}$ |
| 55.2208° | $2.000 \times 10^{-10}$ | $3.001 \times 10^{-7}$ | $2.667 \times 10^{-7}$ | $1.017 \times 10^{-5}$ |

At the standard parallel itself the optimized sphere's $\varepsilon^2$ is indistinguishable from floating-point zero, as expected from a sphere built to hold second-order tangency there, while every traditional sphere sits three to seven orders of magnitude higher throughout the same band; at the band edges, where the optimized sphere is by construction exactly at the $2 \times 10^{-10}$ acceptance threshold of Section 76.4.1, the traditional rectifying and authalic spheres still exceed it by roughly four orders of magnitude, and the traditional conformal sphere by roughly five.

**Key Result**

Independently reproducing the three traditional spheres from their own defining equations confirms, quantitatively rather than only by name, what "low-distortion" means in Kerkovits (2026)'s title: across the entire $\Phi_0 = 45°$ band, the optimized sphere's own distortion is three to thirty orders of magnitude smaller than any of the three classical alternatives evaluated at the same points, with the gap widening away from $\Phi_0$ itself, most sharply for the conformal sphere.

As an aside, not claimed by Kerkovits (2026) and checked here only out of curiosity: scanning each traditional sphere's own $\varepsilon^2$ over the full $-90°$ to $90°$ range shows it is *not* minimized

near $\Phi_0 = 45°$ at all. The traditional conformal sphere's own minimum sits at the equator ($\Phi \approx 0°$, consistent with $R = a$ being the equatorial radius), while the traditional rectifying and authalic spheres are both minimized near the pole ($\Phi \approx 89.9°$ and $\Phi \approx 89.8°$ respectively). This is a restatement, not a new finding: each traditional sphere was built to hold some other global property (total meridian length, total surface area, angle at the equator) rather than to minimize local distortion at any particular mid-latitude parallel, so their comparatively poor showing in Table 76.2 is exactly what each sphere's own construction would predict, not a deficiency uncovered by this comparison.

### 76.4.4 The angular counterpart: Tissot $\omega$ for all four spheres

**Methodological Note**

Chapter 85's closing todobox named this item without attempting it: a matching angular figure for the four spheres of Table 76.2, "pulled from the same $h, k$ values by [Kerkovits's] own indicatrix formula." This subsection does exactly that, using only $h, k$ values already computed by `code/kerkovits_sphere_reproduction.py` and `code/kerkovits_vs_traditional_spheres.py` – no new reproduction, only a new statistic drawn from figures already verified. Since all four spheres map WGS84 meridians and parallels to meridians and parallels of a sphere (no shear: the meridian and parallel directions are the mapping's principal directions throughout), Tissot's semi-axes reduce to $a = \max(h, k)$, $b = \min(h, k)$ and the standard indicatrix opening angle is $\omega = 2\arcsin\left(\frac{a-b}{a+b}\right)$ – the same relation Chapter 31's general Jacobian formula already reduces to whenever its off-diagonal terms vanish, not a new formula introduced here. Source: `code/kerkovits_indicatrix_omega.py`, `data/kerkovits_indicatrix_omega_results.json`.

Every $\varepsilon^2 = \ln^2 h + \ln^2 k$ recomputed by this subsection's script agrees with Table 76.2's already-published values to within $3 \times 10^{-4}$ relative error at all five points and all four spheres (the residual is consistent with that table's own 4-significant-figure rounding), confirming this script has not silently drifted from the formulas it reuses before its new $\omega$ column is taken at face value.

Table 76.3: Tissot angular distortion $\omega$, derived from the same $h, k$ values underlying Table 76.2. Source: `code/kerkovits_indicatrix_omega.py`, `data/kerkovits_indicatrix_omega_results.json`.

| $\Phi$ | optimized | traditional rectifying | traditional authalic | traditional conformal |
|---|---|---|---|---|
| 34.0988° | $8.10 \times 10^{-4}$° | 0.06601° | 0.08803° | 0° |
| 40° | $8.15 \times 10^{-5}$° | 0.05653° | 0.07537° | 0° |
| 45° | $1.27 \times 10^{-14}$° | 0.04819° | 0.06425° | 0° |
| 50° | $8.92 \times 10^{-5}$° | 0.03984° | 0.05312° | 0° |
| 55.2208° | $8.10 \times 10^{-4}$° | 0.03139° | 0.04185° | 0° |

The traditional conformal sphere's $\omega$ is exactly zero at every point tested, to floating-point precision – not a small residual, an identity. This is not a new finding about this particular sphere; it is what "conformal" means, restated numerically: a conformal latitude construction has $h = k$ by definition, so its indicatrix is a circle everywhere, and Table 76.2's comparatively large $\varepsilon^2$ for this sphere (the largest of the four at every point) is therefore *entirely* a scale-

distortion effect ($h = k \neq 1$), with no shape-distortion component at all – a distinction $\varepsilon^2$ alone cannot show, since it sums $\ln^2 h$ and $\ln^2 k$ without separating the isotropic part from the anisotropic one. The optimized sphere's own $\omega$ is likewise driven to near-zero (below $10^{-3\circ}$ throughout, versus the traditional rectifying and authalic spheres' roughly $0.03^\circ$–$0.09^\circ$), but for a different reason: $h \equiv 1$ by construction and $k$ stays within a few parts in $10^4$ of 1 across the whole band (Table 76.1), so $a$ and $b$ are both already close to 1 and close to each other, not because the mapping is conformal in the traditional sense (it is not: $h \neq k$ in general, just barely).

**Key Result**

On the shape axis, ranked from most to least angle-preserving across this band: traditional conformal ($\omega \equiv 0$, by definition) < optimized Kerkovits (2026) sphere ($\omega < 10^{-3\circ}$, by near-tangency of $h$ and $k$ to 1, not by conformality) < traditional rectifying ($0.031^\circ$–$0.066^\circ$) < traditional authalic ($0.042^\circ$–$0.088^\circ$, the largest of the four at every point tested). This is the same three-to-thirty-orders-of-magnitude gap Table 76.2 already reported for $\varepsilon^2$, now decomposed onto its angular axis specifically: the optimized sphere's advantage there is not because it is more conformal than the alternatives (the traditional conformal sphere trivially wins that axis by definition), but because it holds *both* $h$ and $k$ near 1 simultaneously, within a purpose-built band, rather than holding one of them exactly at the cost of the other drifting freely across the same band.

**Methodological Note**

**Whose construction this is.** Kerkovits (2026)'s closing discussion only *suggests*, in one sentence and with no formulae, that using three standard parallels instead of one could widen the low-distortion band at the cost of more complex expressions. What follows is therefore not a reproduction of anything in the cited paper — it is an original construction by this volume, one reasonable way to realize that suggestion and not the only possible one, worked out and verified from scratch in `code/kerkovits_three_parallel_extension.py`. It should be read with the same caution as any new derivation in this volume: checked against a limiting case below, but not benchmarked against any published three-parallel construction, since none is cited or known to exist for this particular sphere.

**Construction.** The rectifying family of Section 76.2 keeps $h \equiv 1$ identically for *any* choice of $(n, R, \kappa)$ once $\varphi(\Phi) = \sigma(\Phi)/R + \kappa$ is imposed; only $k(\Phi) = Rn \cos\varphi(\Phi)/(N(\Phi)\cos\Phi)$ depends on which choice is made. Kerkovits's own construction spends its three degrees of freedom on second-order tangency of $k$ to 1 at a single parallel $\Phi_0$. The natural three-parallel generalization keeps the same three unknowns and the same three degrees of freedom, but spends them on zeroth-order matching at three distinct parallels instead:

$$k(\Phi_1) = k(\Phi_2) = k(\Phi_3) = 1, \tag{76.8}$$

a transcendental system in $(n, R, \kappa)$ with no closed form found here, solved instead by numerical root-finding (`scipy.optimize.fsolve`). The particular placement chosen is the symmetric one, $\Phi_1 = \Phi_0 - \delta$, $\Phi_2 = \Phi_0$, $\Phi_3 = \Phi_0 + \delta$, for a half-spacing $\delta$ free to be tuned.

**Key Result**

As $\delta \to 0$, Eq. (76.8)'s three zeroth-order conditions at three converging points reduce, by symmetric finite-difference construction, to exactly Kerkovits's own value-plus-two-derivative tangency condition at $\Phi_0$ alone. The root-finder was checked against this limit

rather than against any external source: at $\delta = 0.001°$ it returns $n = 1.000842082541$, $R = 6{,}378{,}101.030$ m, $\kappa = 0.002985268413$ rad, matching Table 76.1's single-parallel values to all digits shown. This is an internal consistency check on the numerical setup, not a claim reproduced from the literature.

Table 76.4: Symmetric three-parallel construction: acceptance-band width ($\varepsilon^2 < 2 \times 10^{-10}$) as a function of the half-spacing $\delta$, against the single-parallel width of 21.1219°. Source: `code/kerkovits_three_parallel_extension.py`, `data/kerkovits_three_parallel_extension_results.json`.

| $\delta$ | band low | band high | width | acceptance region |
|---|---|---|---|---|
| 1° | 34.0671° | 55.2524° | 21.1853° | simple |
| 4° | 33.5918° | 55.7270° | 22.1352° | simple |
| 7° | 32.5589° | 56.7653° | 24.2064° | simple |
| 10° | 31.0147° | 58.3291° | 27.3143° | simple |
| 12° | 29.7435° | 59.6214° | 29.8779° | simple |
| 14° | 28.3160° | 61.0744° | 32.7584° | simple |
| 14.2° | — | — | **33.0613°** | simple (widest found) |
| 14.3° | — | — | 25.0390° (naive) | **no longer simple** |
| 15° | 27.5536° | 51.1715° | 23.6179° (naive) | no longer simple |

The band widens with $\delta$ across Table 76.4, from 21.1219° at $\delta = 0$ to a maximum of 33.0613° at $\delta = 14.2°$ — a genuine 56% widening over the single-parallel construction, achieved here entirely by re-spending the same three degrees of freedom differently, with no fourth parameter introduced.

**Methodological Note**

**A complication, not smoothed over.** The widening is not unbounded and not always well-described by a single band width. Beyond $\delta \approx 14.2°$–$14.3°$, $k(\Phi)$ develops enough structure between the three enforced zeros that $\varepsilon^2$ crosses the $2 \times 10^{-10}$ threshold more than twice: at $\delta = 15°$, for instance, a narrow excursion above threshold opens up around $\Phi \approx 51$–$52°$, then closes again before $\Phi_0 + \delta = 60°$, so the "acceptance region" is two disjoint intervals rather than one. The naive first-crossing width reported for $\delta \geq 14.3°$ in Table 76.4 is therefore not a band width in the sense used elsewhere in this chapter, and is marked as such rather than quietly taken at face value. This was checked, not assumed: every entry in the table was tested for exactly how many threshold crossings $\varepsilon^2$ makes on each side of $\Phi_0$, and the bifurcation was localized on a $0.1°$ grid in $\delta$.

**Key Result**

For the symmetric zeroth-order three-parallel construction Eq. (76.8), the acceptance band can be widened from Kerkovits's single-parallel 21.1219° to at most 33.0613° ($\delta \approx 14.2°$) while remaining a single contiguous interval; pushing $\delta$ further trades that simplicity away, opening a disjoint low-distortion pocket instead of a wider single band. This confirms, with an actual number rather than only the paper's own qualitative remark, that a three-parallel extension can widen the band — while also showing, as an original finding of this volume rather than anything claimed by Kerkovits (2026), that the widening is bounded and the simple-band regime has a definite edge.

**To Complete / Verify**

Not attempted here: any placement of the three standard parallels other than the symmetric one (asymmetric spacing, or the two-standard-parallel analogue used for conic map projections elsewhere in this volume's Monograph 4 companion series); first-order (tangency-type) conditions at more than one parallel, which would require more than three free parameters and so a different family than Eq. (76.3); and any comparison against Kerkovits (2026)'s own text, since the paper does not carry out this construction and so provides nothing to check against beyond the $\delta \to 0$ limit already used above. Left open, in the same spirit as Section 114.5's disclosed list.

*Update:* Chapter 77 has since closed the asymmetric-placement item, finding two distinct regimes (symmetric split width-*minimizing* for sub-critical spans; mild asymmetry width-*maximizing* near the bifurcation, extending the best simply-connected width to $33.7055°$ against this chapter's own symmetric maximum of $33.0613°$) rather than a single answer. The first-order-tangency and paper-comparison items remain open.

Chapter 77

# Asymmetric Placement of the Three Standard Parallels

## 77.1 Scope: the first item of Chapter 76's closing todobox

**Methodological Note**

Chapter 76's closing todobox left open, among other items, "any placement of the three standard parallels other than the symmetric one." As with the symmetric construction itself (Section 76.3 onward), this is not anything Kerkovits (2026) works out — the paper's own closing discussion suggests a three-parallel idea in one sentence, symmetric or otherwise, with no formulae. What follows is therefore, again, an original extension by this volume of its own prior construction (`code/kerkovits_three_parallel_extension.py`), not a reproduction of anything published, worked out and checked from scratch in `code/kerkovits_three_parallel_asymmetric.py`. It inherits the same caveat as its predecessor: checked against a limiting case and against Chapter 76's own symmetric results, but benchmarked against no published asymmetric construction, since none is cited or known to exist for this sphere.

## 77.2 Construction and cross-check

The three unknowns and the zeroth-order system are unchanged from Eq. (76.8); only the placement rule changes. Write the three parallels as

$$\Phi_1 = \Phi_0 - \delta_1, \qquad \Phi_2 = \Phi_0, \qquad \Phi_3 = \Phi_0 + \delta_2, \tag{77.1}$$

with total span $S = \delta_1 + \delta_2$ and split fraction $t = \delta_1/S$, so that $t = 1/2$ recovers Chapter 76's symmetric case exactly. $\Phi_0 = 45^\circ$ throughout, matching that chapter. For each fixed $S$, $t$ is swept over a grid in $(0, 1)$ and Eq. (76.8) is re-solved by continuation (each $t$ seeded from the nearest already-converged neighbour), exactly as Chapter 76 did for $\delta$.

**Key Result**

Before using this solver for anything new, $t = 1/2$ was checked at every $S$ already tabulated in Table 76.4 against the independently stored values from that chapter's own run: the largest discrepancy across all seven cross-check points is $3.7 \times 10^{-10\circ}$ in band width, at the level of the root-finder's own tolerance rather than a modelling difference. The asymmetric solver is therefore trusted to reproduce the symmetric one exactly at $t = 1/2$, as it must, before being read for $t \neq 1/2$.

## 77.3 Does asymmetry help, hurt, or neither?

Table 77.1: Acceptance-band width as a function of split fraction $t$, for several fixed total spans $S = \delta_1 + \delta_2$. $t = 0.50$ is the symmetric case; † marks spans where the acceptance region is no longer a single simply-connected interval for that split (naive first-crossing width shown, not a true band width, per Chapter 76's own disclosure). Source: `code/kerkovits_three_parallel_asymmetric.py`, `data/kerkovits_three_parallel_asymmetric_results.json`.

| $S$ | width at $t = 0.05$ | width at $t = 0.50$ | width at $t = 0.95$ | note |
|---|---|---|---|---|
| $10°$ | $23.1408°$ | $22.7033°$ | $23.1690°$ | U-shaped in $t$, minimum at $t$=0.5 |
| $15°$ | $25.5973°$ | $24.6561°$ | $25.6056°$ | same shape |
| $20°$ | $28.8066°$ | $27.3143°$ | $28.7753°$ | same shape |
| $24°$ | $16.2550°$† | $29.8779°$ | $15.7575°$† | simple only for $t \in [0.35, 0.70]$ |
| $28.4°$ | $14.7003°$† | **33.0613°** | $13.8328°$† | simple only for $t \in [0.50, 0.55]$ |

Two distinct regimes emerge, not one. For spans well below the symmetric bifurcation ($S \lesssim 22°$, Table 77.1's first three rows), the acceptance region stays a single simply-connected interval across the *entire* $t$-grid tested, $t \in [0.05, 0.95]$, and its width is a U-shaped function of $t$ with its *minimum*, not maximum, at the symmetric split. A finer sweep at $S = 20°$ pushing $t$ toward 0 and 1 (down to $\delta_2 \approx 0.001°$) shows the width keeps growing monotonically on the $t \to 1$ side, flattening out toward a plateau near $29.14°$ as $\delta_2 \to 0$, some $1.8°$ above the symmetric value of $27.31°$; on the $t \to 0$ side the same growth is interrupted below $t \approx 0.018$–$0.020$, where the region abruptly bifurcates (naive width drops to $\approx 19.6$–$20.5°$) — an asymmetry between the two tails of $t$ that Eq. (77.1)'s construction does not obviously predict, since $M$ and $N$ are even functions of $\Phi$ but not translation-invariant away from the equator, and is reported here as an observed, not derived, fact.

For spans at or beyond Chapter 76's own symmetric bifurcation ($\delta \approx 14.2$–$14.3°$, i.e. $S \approx 28.4$–$28.6°$), the picture reverses: only a narrow window of splits close to $t = 0.5$ remains simply connected at all (Table 77.1's last two rows), and mild asymmetry *within* that window can push the width slightly past the symmetric value ($33.0759°$ at $t = 0.55$, $S = 28.4°$, against $33.0613°$ at $t = 0.5$) — but a symmetry purely local to the vicinity of $t = 0.5$, not a re-run of the U-shape seen at smaller $S$.

**Key Result**

Two findings, of different character. First: for spans comfortably inside the symmetric single-band regime, the symmetric split is *not* width-optimal — it is the width-*minimizing* choice among simply-connected splits tested, and pushing $\delta_2 \to 0$ (a near-degenerate placement with two of the three parallels almost coincident) widens the band by roughly $1.8°$ over the symmetric value at $S = 20°$, a genuinely counter-intuitive result not suggested anywhere in Kerkovits (2026) or in this volume's own prior symmetric chapter. Second, and separately: searching $(S, t)$ jointly (Table 77.2) finds a simply-connected acceptance region as far out as $S = 29.24°$ at $t = 0.52$, giving a width of $33.7055°$ over the interval $[27.3275°, 61.0330°]$ — wider than any symmetric construction achieves, including Chapter 76's own maximum of $33.0613°$ over $[28.1659°, 61.2272°]$ at $\delta = 14.2°$. This closes the todobox item with an actual number, not only a qualitative "asymmetry might help": it both can and cannot, depending on which of the two regimes above the span falls in.

Table 77.2: Localizing the largest total span $S$ for which some split $t$ keeps the acceptance region simply connected, on a $0.01°$ grid in $S$ (having first located the $0.1°$-scale window on a coarser grid, in the same style Chapter 76 used for its own symmetric bifurcation). Source: as Table 77.1.

| $S$ | best $t$ | best width | status |
|---|---|---|---|
| $28.40°$ (symmetric bifurcation) | $0.54$ | $33.0693°$ | simple |
| $29.00°$ | $0.52$ | $33.5205°$ | simple |
| $29.20°$ | $0.52$ | $33.6746°$ | simple |
| $29.24°$ | $0.52$ | $\mathbf{33.7055}°$ | simple (widest found) |
| $29.25°$ | — | — | no longer simple on this $t$-grid |
| $29.30°$ | — | — | no longer simple on this $t$-grid |

**Methodological Note**

**What this table does and does not show.** The $t$-grid used to localize $S = 29.24°$ is $0.01$ wide near $t \approx 0.52$; a finer grid might find a marginally larger simply-connected span or a marginally wider band, so $29.24°/33.7055°$ should be read as the best *found* on this search, not a proven supremum. No attempt is made here to derive the asymmetric bifurcation analytically (the symmetric one was not derived analytically either, in Chapter 76); both remain numerically located boundaries.

**To Complete / Verify**

Left open by this chapter: an analytic account of why $\delta_2 \to 0$ widens the band while $\delta_1 \to 0$ instead bifurcates it at $S = 20°$, rather than the two tails behaving symmetrically; whether the near-degenerate $\delta_2 \to 0$ limit (two of the three zeroth-order conditions coalescing at $\Phi_0$ into what becomes, in the limit, a first-order tangency condition there) is itself a distinct, nameable construction rather than merely a limit of this one; a joint optimization over $(\Phi_0, \delta_1, \delta_2)$ rather than the $\Phi_0 = 45°$ used throughout this volume's Kerkovits chapters; and the two items Chapter 76's own todobox already disclosed and this chapter does not touch — first-order conditions at more than one parallel (a different, four-or-more-parameter family), and comparison against Kerkovits (2026)'s own text, which carries out neither the symmetric nor the asymmetric three-parallel construction.

Chapter 78

# Regional Versus Global: Where Does Local Optimization Win?

## 78.1 The research question

Chapter 75 showed that Kerkovits (2026)'s regional auxiliary sphere buys near-zero linear distortion by spending its entire budget within a fixed latitude band, at the cost of having no defined behaviour outside it. This raises the question left open at the end of that chapter: for which spatial extents does regional optimization actually outperform a global compromise, and where does it fail by extrapolation beyond its band of validity — the same question Ch. 37 of Monograph 4 asked of the empirical IDI regularity, now asked of this different, band-restricted construction.

A full answer would require reproducing Kerkovits (2026)'s sphere numerically and comparing it directly against a specific global DGGS at matched resolution; that is not attempted here (see the Scope: what this result does and does not show section below). What *can* be established rigorously, using only the continuous limit already derived in Chapter 1, is a narrower but still informative result: how much the "cost of regionality" — the internal area variation that a fixed-width regional band still carries — itself depends on where that band is centred.

## 78.2 The cost of a fixed-width band is not uniform in latitude

**Definition 78.1** (Band area-ratio spread)**.** For a latitude band $[\phi_c - w, \phi_c + w]$ of half-width $w$ centred at $\phi_c$, define the area-ratio spread under the continuous limit law $A(\phi) \propto \cos\phi$ of Remark 1.3 as

$$\text{spread}(\phi_c, w) \;:=\; \frac{\cos(\phi_c - w)}{\cos(\phi_c + w)} \;\geq\; 1, \tag{78.1}$$

the ratio between the largest and smallest physical cell area found anywhere within the band.

**Proposition 78.2** (Monotonicity of the band spread)**.** *For fixed $w \in (0, \pi/2)$, $\text{spread}(\phi_c, w)$ is strictly increasing in $\phi_c$ on $\phi_c \in [0, \pi/2 - w)$.*

*Proof.* Write $f(\phi_c) = \ln\cos(\phi_c - w) - \ln\cos(\phi_c + w)$, so spread $= e^{f(\phi_c)}$ and monotonicity of the spread is equivalent to monotonicity of $f$. Then $f'(\phi_c) = \tan(\phi_c + w) - \tan(\phi_c - w)$. Since tan is strictly increasing on $(-\pi/2, \pi/2)$ and $\phi_c + w > \phi_c - w$ throughout the stated range, $f'(\phi_c) > 0$, so $f$, and hence the spread, is strictly increasing in $\phi_c$. □

**Key Result**

A latitude band of fixed angular half-width $w$ carries strictly more internal cell-area variation the closer it sits to the pole (Proposition 78.2). Kerkovits's near-20° band therefore does not cost the same everywhere: centred at the equator, a 20° band is close to homoge-

neous; centred at high latitude, the same angular width already spans a substantial area ratio under the continuous limit law. This is a structural, model-independent reason to expect regional optimization to be more valuable at high latitude than at low latitude — exactly where Chapter 58's boundary law showed the global discrete graticule degenerates fastest.

Table 78.1 makes Proposition 78.2 quantitative for a 20°-wide band (half-width $w = 10°$, matching Kerkovits (2026)'s reported band) centred at ten latitudes from the equator to 75°.

Table 78.1: Area-ratio spread Eq. (78.1) of a 20°-wide band ($w = 10°$) under the continuous limit law, as a function of the band's central latitude $\phi_c$. Source: `code/band_extent_comparison.py`, `data/band_extent_comparison_results.json`.

| $\phi_c$ | band | spread |
|---|---|---|
| 0° | $[-10°, 10°]$ | 1.00000 |
| 10° | $[0°, 20°]$ | 1.06418 |
| 20° | $[10°, 30°]$ | 1.13716 |
| 30° | $[20°, 40°]$ | 1.22668 |
| 40° | $[30°, 50°]$ | 1.34730 |
| 50° | $[40°, 60°]$ | 1.53209 |
| 60° | $[50°, 70°]$ | 1.87939 |
| 65° | $[55°, 75°]$ | 2.21613 |
| 70° | $[60°, 80°]$ | 2.87939 |
| 75° | $[65°, 85°]$ | 4.84900 |

Between the equatorial and the 75°-centred band, the spread grows by very nearly a factor of five, entirely from the fixed angular width being carried across an increasingly compressed range of physical area — the same $\cos\phi$ mechanism responsible for Proposition 58.1's boundary law, now read off over a band rather than a single ring.

*Remark* 78.3 (Relation to the discrete boundary law). Proposition 58.1 and Proposition 78.2 are two views of the same underlying fact: locally, $\cos\phi$ has a derivative $-\sin\phi$ that grows in magnitude with $\phi$, so both a single ring's degeneracy (as $\phi \to \pi/2$, Chapter 58) and a finite band's internal spread (Proposition 78.2, this chapter) worsen toward the pole. The discrete-graticule chapter isolated the limiting, zero-width case; this chapter isolates the finite-width, Kerkovits-relevant case.

## 78.3 Scope: what this result does and does not show

**To Complete / Verify**

Proposition 78.2 and Table 78.1 establish only that a fixed-width band's *internal* homogeneity worsens toward the pole under the continuous limit law — they do not, by themselves, show that Kerkovits (2026)'s specific rectifying sphere construction outperforms a specific global DGGS at any given latitude, since that comparison would require the sphere's actual distortion field (not yet reproduced here; see Chapter 75) set against a chosen global DGGS's actual cell geometry (not yet reproduced either; see Section 1.3). What can honestly be said at this stage is the narrower, but still real, structural point stated in the Key Result above: the potential *gain* available to regional optimization grows with latitude, which is a necessary condition for regional optimization to be worth its cost at high latitude, not a sufficient one. The full comparison, including an explicit diagnosis of where

regional optimization fails by extrapolation beyond its band (in direct analogy with Ch. 37 of Monograph 4), is left for future work, once both reproductions above are complete. Both reproductions are now complete (Chapter 76 for the sphere, Chapters 95 and 97 already complete for two of the five DGGS), and Chapter 80 takes a first, deliberately partial step using them – deliberately partial because, as that chapter explains, the sphere's and the DGGS's distortion figures measure different steps of two different pipelines and are not simply summable into the single comparison this todobox originally envisioned.

Chapter 79

# A Documented National Case: New Zealand and the Price of a Global Auxiliary Sphere

## 79.1 Why this case, and what it does and does not share with Chapter 75

**Methodological Note**

**Independence declared up front.** This chapter is a self-contained addition to Part II, computed from scratch with the libraries already used elsewhere in this volume. It does *not* claim to follow the same protocol as Chapter 75's treatment of Kerkovits (2026) – that chapter's own working files were not available when this one was written, and rather than approximate a method not directly in view, this chapter takes a different, fully independent route to the same question Part II asks: what does a region actually buy, and give up, by using a low-distortion surface built around itself instead of a single global one? Any relationship between the two chapters' findings is for the reader to draw, not asserted here.

New Zealand is not chosen for local interest – it is a documented, independently published instance of exactly this volume's Part II question, decided by a national mapping agency for reasons stated in its own technical literature, not inferred here. Blick & Donnelly (2016) record that New Zealand replaced New Zealand Map Grid (NZMG) with New Zealand Transverse Mercator 2000 (NZTM2000) in 2001, following the 1998 move to the geocentric New Zealand Geodetic Datum 2000 (NZGD2000) built on the GRS80 ellipsoid. Land Information New Zealand's own current guidance is explicit about why NZMG itself had been adopted in the first place: *"NZMG was not a geometric projection…Instead, it used a mathematical expression which combined complex numbers and polynomial terms. This meant that the formula could be tailored to minimalize distortion over New Zealand, but it was unique to New Zealand and could be difficult to include in computer software and navigation equipment."*[1] That sentence states, in a national agency's own words and thirty years before this volume's DGGS framing existed, precisely the trade-off Part II is built around: a bespoke local construction buys distortion reduction at the cost of global interoperability. NZTM2000, its geometric (transverse Mercator) successor, keeps most of the distortion-reduction and restores interoperability with ordinary projection software – at a cost of its own, quantified below.

[1] Land Information New Zealand, "New Zealand Map Grid (NZMG)," Geodetic Guidance, https://www.linz.govt.nz/guidance/geodetic-system/coordinate-systems-used-new-zealand/projections/new-zealand-map-grid-nzmg.

## 79.2 Method

**Methodological Note**

**What was computed.** Seven real New Zealand locations were fixed by their ordinary geographic coordinates (Cape Reinga, Auckland, Wellington, Christchurch, Dunedin, Bluff, and the Chatham Islands – chosen to span the country's full latitude range and, with the Chathams, to include a point known from LINZ's own guidance to sit outside NZTM2000's intended domain). At each point this chapter computed the Tissot indicatrix – $a$, $b$, $\omega$, and the area scale factor – for three maps, using the same general finite-difference construction of Snyder (1987) already used in Chapter 31 (`tissot_from_jacobian()`, reused unchanged from `code/tissot_s2_rhealpix.py`):

- S2's own composite gnomonic-then-quadratic forward map (`s2sphere`), exactly as in Section 31.4;
- rHEALPix's own forward map (`rhealpixdggs`), exactly as in Section 31.3;
- NZTM2000 (EPSG:2193), read from `pyproj`/PROJ's own authoritative definition rather than retyped from a document: `+proj=tmerc +lat_0=0 +lon_0=173 +k=0.9996 +x_0=1600000 +y_0=10000000 +ellps=GRS80`. Because NZTM2000 is defined on a real ellipsoid (GRS80) rather than the unit sphere used for S2 and rHEALPix elsewhere in this volume, its finite-difference Jacobian (in metres per radian) is converted to a dimensionless point scale factor using the local meridional radius of curvature $M$ and prime-vertical radius of curvature $N$ at each point's latitude, rather than dividing by a fixed $R$; the underlying Tissot construction is otherwise identical.

Step-size stability was checked at Wellington across $\varepsilon \in \{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ degrees for NZTM2000 (S2's and rHEALPix's stability at this $\varepsilon$ were already established in Chapter 31); $\omega$ stayed at $10^{-6\circ}$ or below at every step size, i.e. zero within the precision of the method, at every step size tested. Full code: `code/tissot_nz_regional.py`; full output: `data/tissot_nz_regional_results.json`, `data/tissot_nz_regional_convergence.json`.

## 79.3 Results: two global grids, one regional one

Table 79.1: Tissot indicatrix at seven New Zealand locations, three maps. $\omega$ in degrees; NZTM2000's "scale" column is the dimensionless point scale factor $k = \sqrt{\text{area scale}}$, the quantity LINZ's own documentation reports for this projection. Source: `code/tissot_nz_regional.py`, `data/tissot_nz_regional_results.json`.

| Location | S2 $\omega$ | rHEALPix $\omega$ | NZTM $\omega$ | NZTM $k$ |
|---|---|---|---|---|
| Cape Reinga (34.4°S) | 12.787 | 12.653 | $\approx 0$ | 0.999611 |
| Auckland (36.8°S) | 14.569 | 16.094 | $\approx 0$ | 0.999904 |
| Wellington (41.3°S) | 14.265 | 23.198 | $\approx 0$ | 0.999872 |
| Christchurch (43.5°S) | 12.322 | 43.447 | $\approx 0$ | 0.999611 |
| Dunedin (45.9°S) | 11.508 | 41.589 | $\approx 0$ | 1.000062 |
| Bluff (46.6°S) | 11.146 | 39.827 | $\approx 0$ | 1.001170 |
| Chatham Is. (44.0°S) | 15.575 | 46.423 | $\approx 0$ | 1.008263 |

**Key Result**

NZTM2000's angular distortion is zero to the precision of the method at every one of the seven points – not approximately small, but indistinguishable from the $10^{-6\circ}$ numerical floor established by the convergence check. This is not a discovery: a transverse Mercator projection is conformal by construction (Snyder (1987)), so $\omega \equiv 0$ everywhere it is defined is a property of the formula, not a finding about New Zealand. What *is* a finding, because it required computing both maps independently rather than taking the conformal property on faith, is the contrast: at the same seven points, S2 shows $\omega$ between $11.1^\circ$ and $15.6^\circ$, and rHEALPix shows $\omega$ between $12.7^\circ$ and $46.4^\circ$ – more than three times its own lowest value in this table, in a country roughly $13^\circ$ of latitude wide. rHEALPix's steep rise past Wellington ($41.3^\circ$S) to Christchurch ($43.5^\circ$S) lines up with the equatorial-belt/polar-cap seam Chapter 31 already located near $\phi \approx 41.8^\circ$ for this exact $N_side = 3$ construction: four of this chapter's seven points (Christchurch, Dunedin, Bluff, and, by symmetry of the belt/cap structure, the Chathams) sit inside or just past that seam, not because New Zealand is special but because its latitude range happens to straddle a boundary this volume had already identified from the grid's own geometry, not from New Zealand's.

**Key Result**

NZTM2000's point scale factor $k$ is not constant, and its one outlier in this table independently confirms a claim already made in LINZ's own current guidance rather than merely repeating it. Land Information New Zealand states that NZTM2000 *"has distortion in its east-west extents. As a result it is not recommended for mapping offshore islands"*[a] and that separate projections are used for them. The six mainland points in Table 79.1, all within roughly $5^\circ$ of NZTM2000's $173^\circ$E central meridian, show $k$ within $0.06\%$ of unity at the extremes (Bluff, $k = 1.001170$) and closer at the centre (Auckland, $k = 0.999904$, nearest the standard parallels where $k = k_0 = 0.9996$ is exact by definition). The Chatham Islands, roughly $10^\circ$ of longitude east of the central meridian and genuinely outside NZTM2000's intended footprint, show $k = 1.008263$ – a $0.83\%$ linear scale distortion, roughly seven times the worst mainland value in this table. This chapter did not assume LINZ's east-west-distortion claim and transcribe a qualitative description of it; it computed the same projection independently, from PROJ's own numeric definition, and the number that came out is large exactly where LINZ's prose says it should be.

[a] Land Information New Zealand, "New Zealand offshore island projections," Geodetic Guidance, https://www.linz.govt.nz/guidance/geodetic-system/coordinate-systems-used-new-zealand/projections/new-zealand-offshore-island-projections.

## 79.4 The actual price of the regional sphere

Part II's opening question was what a region buys and gives up by building its own low-distortion surface instead of using a single global one. This chapter's numbers answer the "buys" half directly: New Zealand's national projection is, by construction, free of the $11$–$46^\circ$ angular distortion either global grid studied in Chapter 31 carries at the same latitudes. The "gives up" half is not a number this chapter computed, but it is not invented either – it is the same LINZ documentation already cited, read for what it says about cost rather than accuracy:

- **No single global sphere.** LINZ's own guidance lists NZTM2000 for the mainland, a separate Lambert conformal projection (NZCS2000) for the continental shelf, and separate transverse Mercator projections again for the offshore islands and the Chatham Is-

lands[2] – a patchwork of reference surfaces stitched at administrative rather than geometric boundaries, the opposite of a DGGS's one seamless hierarchy referred to throughout Part I.

- **A retired precursor, not a free upgrade.** LINZ's own account of why NZMG – the *more* locally optimized of the two national systems, per the passage quoted in Section 79.1 – was retired in favour of the geometrically plainer NZTM2000 is explicit: a bespoke, non-geometric formula "could be difficult to include in computer software and navigation equipment." The regional sphere with the lowest distortion was not the one New Zealand kept using.
- **A cadastral layer underneath the mapping layer.** Separately from NZTM2000, cadastral surveying in New Zealand still runs on 28 meridional circuits, each its own transverse Mercator instance, because a single national projection was judged insufficiently precise for that use[3]. One country, for one broad purpose set (topographic mapping vs. cadastral survey), already runs more than one regional auxiliary sphere.

None of this is a criticism of NZTM2000 or of New Zealand's geodetic choices, which are well documented and evidently deliberate. It is, instead, an existence proof, drawn from outside this volume's own programme rather than constructed to fit it, that the trade-off Part II poses in the abstract is one a real national mapping agency has faced, weighed, and left a paper trail about.

**To Complete / Verify**

This chapter computed indicatrices at seven point locations, not a continuous distortion field over New Zealand's full extent, and did not attempt the reverse comparison – how a DGGS cell tessellation actually laid over New Zealand would look, cell by cell, against NZTM2000's smooth field. **Resolved by Chapter 90:** this chapter's numbers have since been related to Chapter 80's and Chapter 78's own findings, via the matched latitude band both chapters turn out to share.

[2]Sources as cited above; also Land Information New Zealand, "Meridional circuits," Geodetic Guidance, https://www.linz.govt.nz/guidance/geodetic-system/coordinate-systems-used-new-zealand/projections/meridional-circuits.

[3]Land Information New Zealand, "Meridional circuits," Geodetic Guidance (as above).

Chapter 80

# Regional Sphere Against Global Grid: A First Matched-Band Reading

## 80.1 Scope: a first, deliberately partial answer

**Methodological Note**

Section 78.3's todobox left the direct comparison between Kerkovits (2026)'s sphere and a specific global DGGS explicitly undone, for want of both reproductions. Both are now available: Chapter 76 reproduced the sphere's own distortion law from its defining equations, and S2 and rHEALPix have each already received independent, from-scratch real-geometry treatment elsewhere in this volume (Chapters 95 and 97). This chapter is the first attempt at the comparison, not the full one: it is narrower than what Section 78.3 originally asked for, and Section 80.2 states plainly why.

## 80.2 Why the two numbers are not simply comparable

The rectifying sphere's linear scale $k(\Phi)$ (Chapter 76) measures exactly one step of a double mapping: WGS84 ellipsoid to auxiliary sphere. It says nothing about any further step – there is no second, discretizing stage in Kerkovits (2026)'s construction for this chapter to measure.

A global DGGS's cell-area non-uniformity, as computed in Chapters 95 and 97, measures the opposite end of a different pipeline: how much a fixed tessellation's individual cells deviate from one another in area, on whatever reference surface that DGGS is natively defined – the unit sphere for S2 (no ellipsoidal step exists to measure), WGS84 for rHEALPix when so configured (`rhealpixdggs.ellipsoids.WGS84_ELLIPSOID`, used throughout this chapter). Neither figure includes the other system's missing step, and the two are not additive into a single "total distortion" for either system. This chapter reports both, side by side, in the same latitude band, without summing or ranking them against a common scale – exactly the discipline Section 114.5 already applied to the geometric-boundary-rate-versus-RMSE comparison that Chapter 71 declined to force.

## 80.3 Method

The band is Section 76.4.1's independently reproduced acceptance interval for $\Phi_0 = 45°$, $[34.0988°, 55.2208°]$ (both hemispheres, by symmetry of the underlying formulae). Within this band:

- **S2** (unit sphere, native definition, no ellipsoid): every cell at a fixed level was enumerated exhaustively via `s2sphere`'s `CellId.begin(level)/.next()` (the same exhaustive-sweep discipline as Chapter 95), its real area taken from `Cell.exact_area()`, and its centroid latitude from `LatLng.from_point(Cell.get_center())`, reusing Chapters 62 and 63's already-verified centroid convention. Levels 4 and 5 (1,536 and 6,144 cells globally) were used, matching Chapter 95's own resolution range.
- **rHEALPix** (WGS84 ellipsoid): every cell at a fixed resolution was enumerated via `RHEALPixDGGS.grid(r)`, its real area from `Cell.area(plane=False)`, and its centroid from `Cell.centroid(plane=False)`, mirroring Chapter 97's own method exactly except for the choice of ellipsoid (WGS84 here, to match Kerkovits (2026)'s own reference ellipsoid, rather than the unit sphere used by default elsewhere in Part III). Resolutions 2 and 3 (486 and 4,374 cells globally) were used.

For each grid and resolution, every cell's area was normalized by that resolution's own global mean cell area, then the subset of cells whose centroid latitude falls inside the band was isolated, and the ratio of that subset's maximum to minimum normalized area was recorded as a single band-uniformity figure – the DGGS analogue, restricted to the same latitudes, of asking how far $k(\Phi)$ strays from 1 within Kerkovits (2026)'s band.

## 80.4 Results

Table 80.1: Within-band normalized-area ratio (max/min), S2 and rHEALPix (WGS84), restricted to cells whose centroid latitude falls in $[34.0988°, 55.2208°]$ (or its Southern mirror). Source: `code/kerkovits_vs_dggs_band_comparison.py`, `data/kerkovits_vs_dggs_band_comparison_results.json`.

| grid | reference surface | total cells | cells in band | area ratio (max/min) |
|---|---|---|---|---|
| S2, level 4 | unit sphere | 1,536 | 456 | 1.593786 |
| S2, level 5 | unit sphere | 6,144 | 1,832 | 1.761795 |
| rHEALPix, res. 2 | WGS84 | 486 | 136 | 1.000000 |
| rHEALPix, res. 3 | WGS84 | 4,374 | 1,168 | 1.000000 |

rHEALPix's ratio of exactly 1.000000 within the band is not a close approach to the rectifying sphere's near-perfect uniformity; it is a mechanical consequence of rHEALPix being an equal-area tessellation by construction, at *every* latitude, not specifically within this band – Chapters 100 and 101 already document that this comes at the cost of shape (angular) distortion that grows toward the poles, a distortion axis this chapter's area-only metric cannot see. S2's ratio, by contrast, is a genuine, latitude-band-specific figure: S2 is not equal-area, and its within-band non-uniformity ($1.59$–$1.76\times$, growing mildly with resolution) reflects real cell-area variation that a user gridding data inside this specific $21°$ band would actually encounter.

**Key Result**

On the one axis this chapter can honestly measure – area uniformity within a matched latitude band – S2's real cells vary by a factor of $1.6$–$1.8\times$ across the band that Kerkovits (2026)'s auxiliary sphere renders uniform to within 1:100,000 in linear scale. rHEALPix's equal-area construction trivially matches or exceeds this on the area axis alone, but by a route – global equal-area tessellation – that is not what the rectifying sphere optimizes for, and the comparison says nothing about which system a given regional mapping task

should actually prefer, since Section 80.2's two pipelines differ in what they leave undiscretized.

## 80.5 Which sphere rHEALPix's area axis should actually be read against

Section 76.4.3 has since independently derived and evaluated, across this same $[34.0988°, 55.2208°]$ band, the traditional (non-optimized) authalic and conformal auxiliary spheres alongside the traditional rectifying one. That chapter reported $\varepsilon^2 = \ln^2 h + \ln^2 k$ for each; revisiting its own $h$ and $k$ values here for a different purpose makes Section 80.4's "by construction, not by band choice" remark about rHEALPix precise rather than only qualitative.

At all five points of Table 76.2, the traditional authalic sphere's own $h$ and $k$ satisfy $hk = 1$ to 9–10 decimal places (limited by that chapter's finite-difference step, not by the construction itself), confirming numerically what its definition already guarantees algebraically: an authalic sphere is area-preserving *everywhere*, not merely within whichever band happens to be under discussion. The traditional conformal sphere shows no such thing – its own $hk$ product drifts from $1.0021$ at $34.0988°$ to $1.0045$ at $55.2208°$, real area distortion at the ellipsoid-to-sphere stage that a conformal-sphere-based regional map would carry even inside this band. Kerkovits's own optimized rectifying sphere sits between the two on this axis by construction (its $h \equiv 1$ everywhere, but $k$ only tangent to 1 at $\Phi_0$, so $hk = k$ drifts exactly as much as Table 76.2 already reported for $k$ alone).

**Key Result**

rHEALPix's within-band area ratio of exactly $1.000000$ (Table 80.1) is now anchored to a specific counterpart at the ellipsoid-to-sphere stage: the traditional authalic sphere, which is the *only* one of the three classical auxiliary spheres compared in Section 76.4.3 that shares rHEALPix's area-exactness property, and shares it unconditionally rather than within a tuned band. This sharpens, without changing, Section 80.4's conclusion: rHEALPix was never in competition with the rectifying sphere on the area axis at all, since the rectifying sphere was never built to hold area exact anywhere except by the same partial, in-band coincidence that gives its $k(\Phi_0) = 1$; the sphere it would need to be measured against on that axis is the authalic one, not the one this chapter's band happens to come from.

**To Complete / Verify**

Left open, consistent with Section 114.5's disclosure discipline: no shape/angular (as opposed to area) distortion metric for rHEALPix or S2 has been computed specifically within this band here, though Chapter 31 provides the Tissot machinery to do so. Section 76.4.3 closed the item this chapter previously listed as unattempted, but only for the *traditional* authalic and conformal spheres, not for Kerkovits (2025)'s own *optimized* authalic construction (still not reproduced anywhere in this volume, per Chapter 76's own disclaimer); an optimized-authalic counterpart to Table 80.1's rHEALPix figures, were Kerkovits (2025)'s derivation ever reproduced, remains a distinct open item from the one just closed. H3 has since been added to this table (Chapter 81), which finds H3's within-band area ratio

the largest of the three grids tested. DGGRID and OpenEAGGR have since been added too (Chapter 82), both builds actually carried out rather than deferred; both land close to rHEALPix's end of the table, an order of magnitude tighter than H3 or S2.

*Update:* Chapter 85 has since closed the shape/angular item for S2 and rHEALPix, finding neither grid angle-preserving anywhere in this band and, notably, rHEALPix – exact on area – showing the *wider* angular spread of the two. H3's own $\omega$, and the optimized-authalic and Kerkovits-indicatrix items, remain open as stated above.

Chapter 81

# Extending the Matched-Band Reading to H3

## 81.1 Scope: closing part of ch:kerkovits-vs-dggs-band's todobox

**Methodological Note**

Chapter 80's closing todobox left three items open: a shape/angular distortion metric within the band (not attempted here either; see Section 81.3), the rectifying sphere's own comparison against authalic and conformal low-distortion spheres (also not attempted here), and the absence of H3, DGGRID, and OpenEAGGR from that chapter's matched-band table, "though each has real-geometry code already available elsewhere in this volume that this chapter's method could be pointed at directly." This chapter closes the H3 part of that third item. DGGRID and OpenEAGGR are not included: both require a from-scratch local build (a compiled DGGRID binary; a compiled `libeaggr.so`) carried out earlier in this volume's own construction and not reproduced here, so extending the comparison to them remains genuinely undone, not merely unattempted for lack of code.

## 81.2 Method

Identical to Section 80.3, substituting H3's own real geometry engine (`h3-py` v4, `cell_area()` and `cell_to_latlng()`, already used and cross-checked in Chapters 12 and 62) for S2's and rHEALPix's. Every H3 cell at a fixed resolution is enumerated exhaustively; each cell's area is normalized by that resolution's own global mean cell area; the subset whose centroid latitude falls in Kerkovits (2026)'s $\Phi_0 = 45°$ band, $[34.0988°, 55.2208°]$ (both hemispheres), is isolated; and the max/min ratio of that subset's normalized areas is recorded. Resolutions 2 and 3 are used – the same H3 resolutions already established throughout this volume's other H3 chapters, chosen for consistency with them rather than for cell-count parity with S2 or rHEALPix, which Chapter 95 already discloses H3 does not share with either grid. Source: `code/kerkovits_vs_dggs_band_h3_extension.py`, `data/kerkovits_vs_dggs_band_h3_extension_results.json`.

## 81.3 Results

H3 shows the largest within-band area ratio of the three real grids now measured – $2.33$–$2.37\times$, above even S2's $1.59$–$1.76\times$ and far above rHEALPix's mechanically trivial $1.000000$. This is not entirely a surprise: Kmoch et al. (2022), as already cited in Section 1.3, group H3 and S2 together as the two schemes with the largest global cell-area variation, against rHEALPix, OpenEAGGR, and DGGRID's ISEA-based area preservation, and Chapter 12's own from-scratch

Table 81.1: Table 80.1 extended with H3. Same band, same normalization convention, same max/min statistic.

| grid | reference surface | total cells | cells in band | area ratio (max/min) |
|---|---|---|---|---|
| S2, level 4 | unit sphere | 1,536 | 456 | 1.593786 |
| S2, level 5 | unit sphere | 6,144 | 1,832 | 1.761795 |
| rHEALPix, res. 2 | WGS84 | 486 | 136 | 1.000000 |
| rHEALPix, res. 3 | WGS84 | 4,374 | 1,168 | 1.000000 |
| H3, res. 2 | WGS84 | 5,882 | 1,538 | 2.328015 |
| H3, res. 3 | WGS84 | 41,162 | 10,804 | 2.367325 |

figure for H3 ($\sigma$(norm. area) $\approx 0.134$ globally) is consistent with H3 being one of the two least area-uniform grids in absolute terms. What *is* new here is the ordering *between* H3 and S2 specifically, within this one band: globally the two are grouped as comparable, but restricted to this $21°$-wide band H3's figure is clearly the larger of the two, not merely comparable to it.

**Key Result**

Restricted to Kerkovits (2026)'s own $21°$-wide band, the three real DGGS now measured rank, from most to least area-uniform: rHEALPix ($1.000000$, by construction) $<$ S2 ($1.59$–$1.76\times$) $<$ H3 ($2.33$–$2.37\times$). This is broadly consistent with Kmoch et al. (2022)'s global finding that H3 and S2 are the two least area-uniform of the five schemes they compared, but sharpens it: within this specific band, H3's variation is not merely comparable to S2's, it is clearly larger, by a factor of roughly $1.3$–$1.5$ depending on resolution.

**To Complete / Verify**

Whether H3's larger within-band figure reflects something specific to this $34$–$55°$ band, or is simply the band-restricted view of H3's already-established larger global variation (Section 1.3 and Chapter 12), is not determined here: that would require repeating this and Chapter 80's measurement at several other band placements and checking whether the H3-over-S2 ordering, and its magnitude, stays roughly constant or instead changes with latitude – not attempted here. The three items left open by Chapter 80 itself (shape/angular metric within the band; comparison against authalic and conformal low-distortion spheres; DGGRID and OpenEAGGR) all remain open after this chapter, joined now by this fourth, H3-specific one.

*Update:* Chapter 85 has since closed the shape/angular item, but for S2 and rHEALPix only – H3's own within-band $\omega$ is explicitly left open by that chapter too, since Chapter 31's Jacobian machinery does not cover H3's icosahedral projection. This chapter's own band-placement question (first paragraph above) is untouched.

*Further update:* Chapter 82 has since closed the DGGRID/OpenEAGGR item too, both builds actually carried out there rather than deferred again. Of the four items this todobox and Chapter 80's original one raised, only the band-placement question and H3's own within-band $\omega$ remain open.

*Final update:* Chapter 86 has since closed H3's own within-band $\omega$ too, via Chapter 36's closed-form per-face gnomonic solution rather than the S2/rHEALPix Jacobian machinery this chapter originally pointed to as insufficient. Of the four items this todobox chain raised, only the band-placement question itself remains open.

Chapter 82

# Extending the Matched-Band Reading to DGGRID and OpenEAGGR

## 82.1 Scope: closing the item left open by Chapter 81

**Methodological Note**

Chapter 81 closed the H3 part of Chapter 80's matched-band item and left DGGRID and OpenEAGGR explicitly open, stating that both "require a from-scratch local build ... carried out earlier in this volume's own construction and not reproduced here." This chapter closes both remaining halves of that item together, by actually carrying out both builds rather than pointing to an earlier one: DGGRID 9.0b compiled from source (`github.com/sahrk/DGGRID`, native CMake build) and OpenEAGGR v2.0 compiled from source (`github.com/riskaware-ltd/open-eaggr`, all 35 C++ files, linked against the repository's own bundled static `libgdal.a`/`libproj.a`). One header (`ogrgeojsonreader.h`) was missing from OpenEAGGR's bundled GDAL include tree; it was fetched from the OSGeo/GDAL v2.0.0 tag on GitHub – the exact release the bundle's own `gdal_version.h` reports – rather than reimplemented, to avoid silently diverging from the GDAL behaviour the rest of the library links against. Both builds' provenance is disclosed here in full, consistent with Section 114.5's discipline: no part of either result below comes from a formula standing in for the real grid engine.

## 82.2 Method

Identical in spirit to Section 81.2, substituting each DGGS's own real geometry engine for H3's:

**DGGRID (ISEA7H).** The compiled `dggrid` executable runs `GENERATE_GRID`, `clip_subset_type WHOLE_EARTH`, `geodetic_densify 0.0`, at resolutions 2, 3, and 4 – the same resolutions already used for this DGGS elsewhere in this volume (Chapters 15 and 105). Cell output in AIGEN (ArcInfo Ungenerate) format gives each cell's real, exact boundary polygon; point output gives DGGRID's own reported centroid for that cell. Area is computed from the real boundary polygon by a geodesic polygon-area formula (`pyproj`, authalic sphere, $R = 6{,}371{,}007.2$ m – the same authalic-sphere convention `h3-py` uses internally), not read off any precomputed formula. Source: `code/kerkovits_vs_dggs_band_dggrid_extension.py`.

**OpenEAGGR (ISEA3H).** Cells are enumerated from the 20 icosahedron-face base cells via `EAGGR_GetDggsCellChildren`, recursively, with de-duplication by cell-ID string (this library's parent-child relation is not a clean tree; see Chapter 16). Each cell's real outline comes from `EAGGR_ConvertDggsCellOutlineToShapeString` (WKT); area is computed by `pyproj`'s WGS84-ellipsoid geodesic formula, the same `pyproj_wgs84` method already reported alongside a spherical-excess cross-check in Chapter 16, kept identical here for direct comparability with that chapter rather than switched to the authalic-sphere convention used for DGGRID

and H3 above. The same disclosed NaN-vertex artifact reported in Chapter 16 recurs here (cells adjacent to the icosahedron's singular vertices); affected cells are skipped, not patched. Resolutions 2 and 4 match that chapter's own choice for this DGGS. Centroid latitude is the mean of each cell's distinct boundary-vertex latitudes – OpenEAGGR's outline call returns no separate centroid point the way DGGRID's point output or `h3.cell_to_latlng()` do, so this is a disclosed approximation, good to well under the band's width for cells this small, not the same "library-reported representative point" standard used for the other two DGGS. Source: `code/kerkovits_vs_dggs_band_eaggr_extension.py`.

Both scripts reuse Section 80.3's band and statistic exactly: Kerkovits (2026)'s $\Phi_0 = 45°$ band, $[34.0988°, 55.2208°]$ (both hemispheres); per-cell area normalized by that resolution's own global mean; max/min ratio of the in-band subset reported.

## 82.3 Results

Table 82.1: Table 81.1 extended with DGGRID and OpenEAGGR. Same band, same normalization convention, same max/min statistic. Note the two DGGS use different geodetic reference conventions for area (authalic sphere for DGGRID, WGS84 ellipsoid for OpenEAGGR, each matching that DGGS's own established chapter elsewhere in this volume); the ratios are dimensionless and comparable across the table despite this, but the absolute cell-area figures underlying them are not on an identical footing.

| grid | reference surface | total cells | cells in band | area ratio (max/min) |
|---|---|---|---|---|
| S2, level 4 | unit sphere | 1,536 | 456 | 1.593786 |
| S2, level 5 | unit sphere | 6,144 | 1,832 | 1.761795 |
| rHEALPix, res. 2 | WGS84 | 486 | 136 | 1.000000 |
| rHEALPix, res. 3 | WGS84 | 4,374 | 1,168 | 1.000000 |
| H3, res. 2 | WGS84 | 5,882 | 1,538 | 2.328015 |
| H3, res. 3 | WGS84 | 41,162 | 10,804 | 2.367325 |
| DGGRID ISEA7H, res. 2 | authalic sphere | 492 | 136 | 1.013667 |
| DGGRID ISEA7H, res. 3 | authalic sphere | 3,432 | 904 | 1.046468 |
| DGGRID ISEA7H, res. 4 | authalic sphere | 24,012 | 6,264 | 1.013534 |
| OpenEAGGR ISEA3H, res. 2 | WGS84 | 560 | 130 | 1.047164 |
| OpenEAGGR ISEA3H, res. 4 | WGS84 | 7,500 | 2,060 | 1.069837 |

Both DGGRID and OpenEAGGR land close to rHEALPix's end of the table, not H3's or S2's: within-band area ratios of roughly $1.01$–$1.05\times$ and $1.05$–$1.07\times$ respectively, an order of magnitude tighter than H3's $2.33$–$2.37\times$ or S2's $1.59$–$1.76\times$, though neither is exactly $1.000000$ the way rHEALPix's construction forces. This is the expected qualitative result: both DGGRID's ISEA7H and OpenEAGGR's ISEA3H are, like rHEALPix, built on Snyder's equal-area icosahedral projection (ISEA), so a small residual band-restricted variation rather than H3's or S2's much larger one is consistent with Kmoch et al. (2022)'s grouping (already invoked in Sections 1.3 and 81.3) of rHEALPix, OpenEAGGR, and DGGRID together as the area-preserving cluster, against H3 and S2 as the two schemes with materially larger area variation. DGGRID's figure does not monotonically shrink with resolution (res. 3's $1.046$ sits above both res. 2's and res. 4's $\approx 1.014$), which is not explained here; it is left as an open question below rather than smoothed into a monotone-looking trend.

**Key Result**

Restricted to Kerkovits (2026)'s own 21°-wide band, all five real DGGS now measured rank, from most to least area-uniform: rHEALPix (1.000000, by construction) < DGGRID (1.01–1.05×) ≈ OpenEAGGR (1.05–1.07×) < S2 (1.59–1.76×) < H3 (2.33–2.37×). This closes, for all five grids compared anywhere in this volume, the ordering Chapter 81's keyresult established for three of them, and confirms it extends cleanly to the two Snyder-ISEA-projection DGGS once their own real geometry is actually built and measured, rather than merely assumed by analogy with rHEALPix.

**To Complete / Verify**

The non-monotonic DGGRID figure across resolutions 2–4 noted above is not explained; it may reflect genuine resolution-dependent structure in ISEA7H's aperture-7 hexagon/pentagon mix within this specific band, or simply this band's particular placement relative to the resolution grid's own great-circle rows, or a coincidence of these three resolutions specifically – distinguishing between these would require measuring additional resolutions, not attempted here. OpenEAGGR's centroid-latitude approximation (mean of boundary vertices, disclosed in Section 82.2) has not been cross-checked against a library-reported representative point, since OpenEAGGR's API exposes none for a cell outline; the size of any resulting error is not quantified here. Chapter 81's own open items – the band-placement question (whether H3's and now DGGRID's/ OpenEAGGR's orderings hold at other latitudes) and the still-missing shape/angular metric and optimized-authalic comparison for H3 – all remain open after this chapter, which closes only the DGGRID/OpenEAGGR presence item itself.

**Update:** Both of those items are since closed. Chapter 86 supplies H3's own within-band $\omega$ from its closed-form per-face gnomonic Tissot solution, and records that the optimized-authalic comparison was already settled separately by Section 76.4.4. Chapter 87 closes the band-placement question, testing two new bands against all five DGGS. This chapter's own two items above (the non-monotonic DGGRID figure, and OpenEAGGR's uncross-checked centroid approximation) remain open; no later chapter in this volume addresses either.

**Update:** Both remain open by design rather than by oversight. Resolving either requires the compiled DGGRID binary and the `eaggr` bindings themselves – to test additional ISEA7H resolutions for the first item, or to query OpenEAGGR's own representative-point output (where exposed) against this volume's boundary-mean approximation for the second – and neither toolchain is part of this volume's software environment (Section 82.2 already discloses the boundary-mean workaround this forced). Building that toolchain from source was judged, on balance, not worth doing for these two items alone: it is a standing cost paid again in any future session rather than a one-time investment, with no guarantee that a successful build would immediately explain the non-monotonicity rather than simply add more measured points to the same open question. Per this volume's disclosed-limitation practice (Section 11.5 and elsewhere), both items are therefore left explicitly open, as described above, rather than closed by a partial or rushed measurement.

Chapter 83

# The Matched-Band Reading Against the Asymmetric, Widened Band

## 83.1 Scope: does a wider band change the ordering?

**Methodological Note**

Chapters 80 to 82 all measure real DGGS cell-area uniformity restricted to Kerkovits (2026)'s own symmetric 21°-wide band, $[34.0988°, 55.2208°]$. Chapter 77 has since found a wider, asymmetric band, $[27.3275°, 61.0330°]$ ($33.71°$ wide, an original construction of this volume rather than anything in the cited paper), for which the auxiliary sphere still holds its own acceptance criterion. This chapter asks a narrow, mechanical question: restricted to that *wider* band instead of the original one, does the DGGS ordering established in Chapter 82's keyresult change? Only S2, rHEALPix, and H3 are re-measured here – DGGRID and OpenEAGGR are not, for the identical reason Chapter 81 gave for its own initial H3-only scope: both require a from-scratch local build not repeated here. Extending this specific, asymmetric-band comparison to them is left open below.

## 83.2 Method

Identical to Sections 80.3 and 81.2 in every respect except the band: each grid's cells are enumerated exhaustively at the same resolutions already used in those two chapters (S2 levels 4–5, rHEALPix resolutions 2–3 on WGS84, H3 resolutions 2–3), each cell's area normalized by that resolution's own global mean, and the max/min ratio taken over the subset whose centroid latitude falls in $[27.3275°, 61.0330°]$ (or its Southern mirror) instead of $[34.0988°, 55.2208°]$. Source: `code/kerkovits_vs_dggs_band_asymmetric_extension.py`, `data/kerkovits_vs_dggs_band_asymmetric_extension_results.json`.

## 83.3 Results

Two of the three grids respond to the wider band, one does not, and each for a disclosed, mechanical reason rather than a coincidence. **rHEALPix** stays at exactly $1.000000$: as Section 80.4 already established, this is a consequence of equal-area construction holding at *every* latitude, not a property of the particular band tested, so widening the band cannot move it. **H3**'s figure is, to six decimal places, identical between the two bands ($2.328015$ and $2.367325$ at res. 2 and 3 respectively, unchanged). This is not the same mechanism as rHEALPix's — H3 is not equal-area – but rather that the specific cells realizing H3's global within-band maximum and minimum area already fall inside the narrower symmetric band, so extending the band outward by roughly $6°$ on each side adds cells without adding any new extremum on either end of the ra-

Table 83.1: Table 81.1's three pip-installable grids, re-measured within Chapter 77's wider, asymmetric band instead of Kerkovits (2026)'s own symmetric one. The symmetric-band column reproduces Tables 81.1 and 82.1 exactly, for direct side-by-side reading.

| grid | reference surface | ratio, symmetric band $(21°)$ | ratio, asymmetric band $(33.71°)$ |
|---|---|---|---|
| S2, level 4 | unit sphere | 1.593786 | 1.723416 |
| S2, level 5 | unit sphere | 1.761795 | 1.885796 |
| rHEALPix, res. 2 | WGS84 | 1.000000 | 1.000000 |
| rHEALPix, res. 3 | WGS84 | 1.000000 | 1.000000 |
| H3, res. 2 | WGS84 | 2.328015 | 2.328015 |
| H3, res. 3 | WGS84 | 2.367325 | 2.367325 |

tio. **S2**, by contrast, genuinely shifts: $1.593786 \to 1.723416$ at level 4 and $1.761795 \to 1.885796$ at level 5, an increase of $0.08$–$0.13$ in the ratio at both resolutions, meaning S2's more extreme cells (by area) lie in the newly included $27.33°$–$34.10°$ and $55.22°$–$61.03°$ strips, not only within the original $34.10°$–$55.22°$ core.

**Key Result**

Restricted to Chapter 77's wider, asymmetric band, the ordering Chapter 82's keyresult established for the symmetric band is unchanged for these three grids: rHEALPix $(1.000000$, unchanged$)$ $<$ S2 $(1.72$–$1.89\times$, up from $1.59$–$1.76\times)$ $<$ H3 $(2.328015$–$2.367325$, unchanged to six decimal places$)$. The wider band does not flip S2 and H3's relative position, and does not disturb rHEALPix's construction-level exactness; its only effect on the three grids measured here is to widen S2's figure, since the wider band happens to reach cells more extreme than any inside the original $21°$ core, while H3's already-extreme cells were captured either way.

**To Complete / Verify**

DGGRID and OpenEAGGR are not re-measured here, for the build-availability reason stated in Section 83.1; whether their own symmetric-band figures $(1.01$–$1.05\times$ and $1.05$–$1.07\times$ respectively, Table 82.1$)$ shift the way S2's did, stay fixed the way H3's did, or move in between, is not known and would require repeating Chapter 82's own from-scratch builds against this chapter's band instead. H3's own within-band shape/angular metric $(\omega)$ remains open, exactly as Chapters 81 and 85 already disclosed for the symmetric band; it is not attempted here for the asymmetric one either. Whether the near-coincidence between H3's two band figures reflects a genuinely stable global extremum location (as suggested above) or is itself an artifact of these two particular resolutions is not checked against a third resolution here.

Chapter 84

# Closing the DGGRID/OpenEAGGR Item for the Asymmetric, Widened Band

## 84.1 Scope: closing the item left open by Chapter 83

**Methodological Note**

Chapter 83 re-measured S2, rHEALPix, and H3 within Chapter 77's wider, asymmetric band, $[27.3275°, 61.0330°]$, and left DGGRID and OpenEAGGR explicitly open, for the same from-scratch-build reason Chapter 81 originally gave and Chapter 82 then closed for the symmetric band. This chapter closes both remaining halves of that item by carrying out both builds again from a fresh clone rather than reusing any earlier binary: DGGRID compiled from source (`github.com/sahrk/DGGRID`, commit `04bf5ed`, native CMake build) and OpenEAGGR v2.0 compiled from source (`github.com/riskaware-ltd/open-eaggr`, all 35 C++ files, linked against the repository's own bundled static `libgdal.a`/`libproj.a`). The same missing header (`ogrgeojsonreader.h`) recurred and was fetched again from the OSGeo/GDAL v2.0.0 tag, the version the bundle's own `gdal_version.h` reports; that header in turn required two further bundled-but-unreferenced GDAL headers (`ogrsf_frmts.h`, `json.h`) to be added to the include path, not copied or modified. DGGRID's fresh `GENERATE_GRID` run reproduced Chapter 82's exact cell counts (492, 3,432, 24,012 at resolutions 2–4) before any band statistic was computed, confirming this independent build's geometry matches the earlier one bit-for-bit at the enumeration level; OpenEAGGR's fresh `libeaggr.so` reports `EAGGR_GetVersion v2.0`, matching every other OpenEAGGR chapter in this volume. Both builds' provenance is disclosed here in full, consistent with Section 114.5's discipline.

## 84.2 Method

Identical to Section 82.2 in every respect except the band: DGGRID (ISEA7H, resolutions 2–4, AIGEN boundary polygons, authalic-sphere geodesic area) and OpenEAGGR (ISEA3H, resolutions 2 and 4, WKT outlines, WGS84-ellipsoid geodesic area, boundary-vertex-mean centroid) are each re-measured with the in-band subset now selected by $[27.3275°, 61.0330°]$ instead of $[34.0988°, 55.2208°]$, matching Section 83.2's band exactly. Sources: `code/kerkovits_vs_dggs_band_dggrid_asymmetric_extension.py`, `code/kerkovits_vs_dggs_band_eaggr_asymmetric_extension.py`.

## 84.3 Results

Table 84.1: Table 83.1 extended with DGGRID and OpenEAGGR, completing the five-grid asymmetric-band reading. The symmetric-band column reproduces Table 82.1 exactly.

| grid | reference surface | ratio, symmetric band ($21°$) | ratio, asymmetric band ($33.71°$) |
|---|---|---|---|
| S2, level 4 | unit sphere | 1.593786 | 1.723416 |
| S2, level 5 | unit sphere | 1.761795 | 1.885796 |
| rHEALPix, res. 2 | WGS84 | 1.000000 | 1.000000 |
| rHEALPix, res. 3 | WGS84 | 1.000000 | 1.000000 |
| H3, res. 2 | WGS84 | 2.328015 | 2.328015 |
| H3, res. 3 | WGS84 | 2.367325 | 2.367325 |
| DGGRID ISEA7H, res. 2 | authalic sphere | 1.013667 | 1.174055 |
| DGGRID ISEA7H, res. 3 | authalic sphere | 1.046468 | 1.388844 |
| DGGRID ISEA7H, res. 4 | authalic sphere | 1.013534 | 1.158234 |
| OpenEAGGR ISEA3H, res. 2 | WGS84 | 1.047164 | 1.047543 |
| OpenEAGGR ISEA3H, res. 4 | WGS84 | 1.069837 | 1.069837 |

The two Snyder-ISEA grids split sharply, and, unlike Chapter 83's S2/H3 pair, their symmetric-band closeness does not survive the wider band. **DGGRID** moves the most of any grid measured across either chapter: $1.3$–$4.2$ percentage points at res. 2 and 4, but a jump of $0.34$ (roughly one third) at res. 3, from $1.046468$ to $1.388844$. This is the same qualitative behaviour S2 showed in Section 83.3 – the widened band reaches genuinely more extreme cells than the original $21°$ core contained – but larger in relative terms than S2's own $0.08$–$0.13$ shift. **OpenEAGGR**, by contrast, behaves like H3: its res. 2 figure moves only in the sixth decimal place ($1.047164 \to 1.047543$), and its res. 4 figure is not merely close but *identical* to six decimal places ($1.069837$ both bands) – and the underlying `area_min`/ `area_max` pair driving that ratio is bit-for-bit the same value in both this chapter's and Chapter 82's result files, confirming the same two cells (not merely a coincidental ratio) realize both the global and the in-band extremum at that resolution, exactly the mechanism Section 83.3 inferred for H3 from ratio-matching alone, here directly confirmed against the raw area values.

**Key Result**

Restricted to Chapter 77's wider, asymmetric band, the complete five-grid ordering is: rHEALPix ($1.000000$, unchanged) $<$ OpenEAGGR ($1.048$–$1.070\times$, essentially unchanged) $<$ DGGRID ($1.158$–$1.389\times$, up sharply from $1.014$–$1.046\times$) $<$ S2 ($1.723$–$1.886\times$) $<$ H3 ($2.328$–$2.367\times$, unchanged). This is *not* the symmetric-band ordering: Chapter 82 found DGGRID ($1.014$–$1.046\times$) uniformly below OpenEAGGR ($1.047$–$1.070\times$) within the original $21°$ band, with no overlap between the two ranges. The wider band inverts that pair – DGGRID's minimum ($1.158$) now exceeds OpenEAGGR's maximum ($1.070$) – while leaving every other grid's relative position from Chapter 82's original five-grid ordering unchanged. Both DGGRID and OpenEAGGR remain, in both bands, closer to rHEALPix's end of the table than to S2's or H3's, consistent with Kmoch et al. (2022)'s area-preserving-cluster grouping of all three Snyder-ISEA-projection grids; what changes is only the two Snyder-ISEA grids' order *relative to each other*, driven entirely by DGGRID's much larger sensitivity to the band's extreme latitudes.

**To Complete / Verify**

DGGRID's non-monotonic pattern across resolutions, already flagged unexplained in Section 82.3 for the symmetric band (res. 3's figure sitting above both res. 2's and res. 4's), recurs here in sharper form: res. 3's asymmetric-band figure ($1.388844$) is not just above but roughly $0.23$–$0.35$ higher than res. 2's and res. 4's ($1.174055$, $1.158234$), a larger relative gap than in the symmetric band. Whether this reflects genuine resolution-dependent structure in ISEA7H's aperture-7 mix, this band's particular placement relative to the resolution grid's rows, or some interaction between the two, is not distinguished here; it would require measuring additional resolutions. OpenEAGGR's centroid-latitude approximation remains uncross-checked against a library-reported representative point, as in Section 82.3. H3's own shape/angular metric ($\omega$) and the band-placement question – whether any of these five orderings hold at latitudes other than Kerkovits (2026)'s and this volume's own two tested bands – remain open for all five grids, exactly as Sections 81.3 and 82.3 already disclosed; nothing in this chapter narrows either question.

Chapter 85

# The Missing Axis: Shape Distortion Within the Same Band

## 85.1 Scope: closing the first item of ch:kerkovits-vs-dggs-band's todobox

**Methodological Note**

Chapter 80's closing todobox, repeated verbatim by Chapter 81, named as its first open item: no shape/angular (as opposed to area) distortion metric for rHEALPix or S2 had been computed specifically within Kerkovits (2026)'s own band, "though Chapter 31 provides the Tissot machinery to do so." This chapter closes that item, and only that one – the rectifying sphere's own comparison against the traditional authalic and conformal spheres was separately closed by Section 80.5, and the H3/DGGRID/OpenEAGGR gaps named by Chapter 81 are untouched here.

## 85.2 Method

Rather than evaluating Tissot's indicatrix at an arbitrary transect, as Chapter 31 did along a face-center-to-corner diagonal, this chapter evaluates it at the *same* real cell centroids already enumerated and area-analyzed by Section 80.3: for each grid and resolution, every in-band cell's centroid (S2: `LatLng.from_point(Cell.get_center())`; rHEALPix: `Cell.centroid(plane=False)` on WGS84, both already verified there) is reused as the evaluation point for Chapter 31's own finite-difference Jacobian-to-$(a, b, \omega)$ machinery (`tissot_from_jacobian`), copied unchanged rather than re-derived. The angular distortion $\omega$ (the full opening angle between the indicatrix's semi-axes, in degrees; $\omega = 0$ is angle-preserving) is computed at every in-band centroid and summarized by its band-restricted minimum, mean, and maximum, exactly mirroring Section 80.3's own min/max treatment of area. A convergence check repeats a handful of these evaluations at four step sizes ($10^{-2}$ to $10^{-5}$ degrees) to confirm $\omega$ is stable well past the fourth significant figure, the same discipline Chapter 31 already applied to its own transect points. Source: `code/kerkovits_vs_dggs_band_shape_metric.py`, `data/kerkovits_vs_dggs_band_shape_metric_results.json`.

Chapter 31's own rHEALPix Jacobian is built on `RHEALPixDGGS(ellipsoid=UNIT_SPHERE)`, not WGS84 – that chapter's own stated scope, not something introduced here. This chapter evaluates that unit-sphere Jacobian at (lon, lat) centroids that were themselves generated by Section 80.3's WGS84-ellipsoid grid, to keep the evaluation points identical to the ones

already reported in Table 80.1. No ellipsoidal correction to the Jacobian is attempted here: the $\omega$ values below measure the unit-sphere rHEALPix map's own angular distortion at those geographic coordinates, not a separately re-derived WGS84-ellipsoid Tissot field. S2 carries no such gap, having no ellipsoidal step at all (Section 80.3). This limitation is inherited, not new, and is disclosed here because this chapter is the first to lean on that Jacobian for a quantitative claim rather than an illustrative transect.

## 85.3 Results

Table 85.1: Tissot angular distortion $\omega$ (degrees), evaluated at every in-band cell centroid already reported in Table 80.1. Same band, same cell population, same resolutions. Source: `code/kerkovits_vs_dggs_band_shape_metric.py`, `data/kerkovits_vs_dggs_band_shape_metric_results.json`.

| grid | reference surface | cells in band | $\omega_{\text{min}}$ | $\omega_{\text{mean}}$ | $\omega_{\text{max}}$ |
|---|---|---|---|---|---|
| S2, level 4 | unit sphere | 456 | $10.7070°$ | $15.6856°$ | $28.0152°$ |
| S2, level 5 | unit sphere | 1,832 | $10.1114°$ | $15.9995°$ | $29.5501°$ |
| rHEALPix, res. 2 | WGS84 centroids* | 136 | $15.7894°$ | $21.9856°$ | $40.1999°$ |
| rHEALPix, res. 3 | WGS84 centroids* | 1,168 | $15.5559°$ | $26.4386°$ | $46.4367°$ |

*Evaluation points from the WGS84 grid; the Jacobian itself is the unit-sphere one, per the integrity box above.

Every figure in this table is strictly positive: neither grid is angle-preserving anywhere in this band, at any resolution tested. This was expected of S2 (a cube-to-sphere gnomonic-family projection has no conformal region away from face centers) but is worth stating plainly for rHEALPix, whose *area* ratio in the same band was exactly $1.000000$ (Table 80.1): rHEALPix's equal-area property buys area uniformity at the direct expense of angular uniformity, not in addition to it. Within this band, rHEALPix's own $\omega$ range ($15.6°$–$46.4°$) is both higher and wider than S2's ($10.1°$–$29.6°$) at the matching resolution pair, and its mean rises measurably from resolution 2 to resolution 3 ($21.99° \rightarrow 26.44°$) while S2's mean is nearly flat ($15.69° \rightarrow 16.00°$) – the opposite of the area axis, where rHEALPix was exact at both resolutions and S2 alone showed resolution-dependence.

**Key Result**

On the shape/angular axis left open by Chapter 80, rHEALPix is *not* the more uniform grid within this band: it is S2 that shows the smaller and more resolution-stable $\omega$ range, while rHEALPix – exact on area – shows the larger and resolution-growing one. Combined with Table 80.1, no grid tested here is uniform on both axes at once inside this band: S2 trades mild area non-uniformity ($1.59$–$1.76\times$) for a comparatively narrower angular spread; rHEALPix trades exact area for the wider one. This is the concrete instance, for this specific band and these two specific grids, of the qualitative point Section 80.2 raised only abstractly: an equal-area DGGS's own distortion axis is angular, not absent, and comparing it to a rectifying sphere – itself built to hold linear scale, not area, uniform – on the area axis alone was always going to favor whichever side of the pipeline happens to be equal-area by construction.

**To Complete / Verify**

This closes Chapter 80's first open item for S2 and rHEALPix specifically; H3's own within-band $\omega$ is not computed here, and would need Chapter 31's Jacobian machinery extended to H3's icosahedral projection first (that chapter covers only S2 and rHEALPix). The traditional and optimized authalic/conformal spheres' own $\omega$ is not addressed here either – Section 80.5 handled their area axis ($\varepsilon^2 = \ln^2 h + \ln^2 k$) only, and a matching angular figure for them would need to be pulled from the same $h, k$ values by Kerkovits (2026)'s own indicatrix formula, not attempted in this chapter. Finally, the unit-sphere-versus-WGS84 gap disclosed above for rHEALPix's Jacobian remains open in exactly the sense Chapter 31 already left it: a genuinely ellipsoidal rHEALPix Tissot field, as opposed to a unit-sphere one evaluated at WGS84-derived points, has not been built anywhere in this volume.

*Update:* Section 76.4.4 has since closed the sphere-side item: the traditional conformal sphere is angle-preserving identically ($\omega \equiv 0$, by definition), the optimized Kerkovits (2026) sphere's own $\omega$ stays below $10^{-3\circ}$ throughout the band, and the traditional rectifying and authalic spheres sit at $0.03°$–$0.09°$, authalic the largest of the four. The H3 and unit-sphere-versus-WGS84 items above remain open.

*Update:* Chapter 86 has since closed the H3 item too, via Chapter 36's closed-form per-face gnomonic solution rather than an extension of this chapter's own Jacobian machinery. Only the unit-sphere-versus-WGS84 rHEALPix gap above remains open.

*Update:* Chapter 33 has since closed the unit-sphere-versus-WGS84 rHEALPix item too, evaluating the library's own ellipsoidal forward transform directly rather than approximating the ellipsoid from the unit-sphere case. The equatorial-belt angular distortion shifts by up to $0.28°$ under the ellipsoid, changing sign near $\phi \approx 22$–$25°$; the polar-cap shift is smaller by roughly two orders of magnitude. All items this todobox originally raised are now closed.

Chapter 86

# H3's Missing Axis: Closing the Angular-Distortion Item

## 86.1 Scope: the last item of a four-chapter chain

**Methodological Note**

Chapter 80's closing todobox named four items; Section 81.3's own "Update" and "Further update" notes tracked three of them closing in turn (Chapter 85: shape/angular metric, S2 and rHEALPix only; Chapter 82: DGGRID and OpenEAGGR presence), leaving exactly two open: the band-placement question, and H3's own within-band angular distortion $\omega$ – explicitly out of reach for Chapter 85 because Chapter 31's Jacobian machinery covers only S2's and rHEALPix's global map definitions, not H3's per-face icosahedral gnomonic one. This chapter closes the second item. It reuses Chapter 36's own closed-form Tissot solution for that projection – derived and cross-checked there against both finite differences and real `h3-py` cell geometry along single transects from two face centers – and applies it, for the first time in this volume, across every real in-band cell centroid rather than an illustrative transect, for *both* bands now established: Kerkovits (2026)'s own symmetric one and Chapter 77's wider, asymmetric one. The band-placement question itself is not addressed here.

## 86.2 Method

Identical cell population to Chapter 81 (`h3.get_res0_cells()` / `h3.cell_to_children()`, resolutions 2 and 3, `h3.cell_to_latlng()` centroids), now filtered against both $[34.0988°, 55.2208°]$ and $[27.3275°, 61.0330°]$. For each in-band centroid, the nearest of the icosahedron's 20 face centers is found by maximum dot product – the identical selection rule H3's own C source uses (`_vec3ToClosestFace`) – and the great-circle angular distance $c$ to it computed. Chapter 36's closed-form result for the per-face azimuthal gnomonic map, $h(c) = \sec^2 c$, $k(c) = \sec c$, $\omega = 2\arcsin\left(\frac{h-k}{h+k}\right)$, is evaluated at that $c$; that chapter's own derivation already established that H3's fixed per-face reference rotation and Class-III offset cannot change $\omega$, only re-orient the indicatrix's axes, so no further per-cell correction is needed. As an additional band-wide check beyond Chapter 36's single-transect one, every 20th in-band cell at resolution 2 in the symmetric band (77 cells) has its closed-form $h/k = \sec c$ compared against the same empirical vertex-distance-ratio proxy that chapter used (`cell_elongation_proxy`, the ratio of the real cell boundary's farthest to nearest centroid-to-vertex great-circle distances); mean absolute difference across those 77 cells is $0.0115$, confirming the closed-form prediction tracks real cell shape band-wide, not only along the two transects already checked. Source: `code/kerkovits_vs_dggs_band_h3_shape_metric.py`, all 20 face centers transcribed from H3's public source exactly as Chapter 36 already did for its own 6-face subset.

## 86.3 Results

Table 86.1: Table 85.1 extended with H3, for both bands. Same $\omega$ definition, same closed-form machinery, same min/mean/max summary discipline.

| grid, band | reference surface | cells in band | $\omega_{\text{min}}$ | $\omega_{\text{mean}}$ | $\omega_{\text{max}}$ |
|---|---|---|---|---|---|
| S2, level 4, symmetric | unit sphere | 456 | 10.7070° | 15.6856° | 28.0152° |
| S2, level 5, symmetric | unit sphere | 1,832 | 10.1114° | 15.9995° | 29.5501° |
| rHEALPix, res. 2, symmetric | WGS84 centroids* | 136 | 15.7894° | 21.9856° | 40.1999° |
| rHEALPix, res. 3, symmetric | WGS84 centroids* | 1,168 | 15.5559° | 26.4386° | 46.4367° |
| H3, res. 2, symmetric | unit sphere | 1,538 | 0.0000° | 3.9123° | 13.1404° |
| H3, res. 3, symmetric | unit sphere | 10,804 | 0.0000° | 3.9088° | 13.1404° |
| H3, res. 2, asymmetric | unit sphere | 2,444 | 0.0000° | 3.8127° | 13.1404° |
| H3, res. 3, asymmetric | unit sphere | 17,151 | 0.0000° | 3.8060° | 13.1404° |

*As in Table 85.1: rHEALPix's Jacobian is the unit-sphere one, evaluated at WGS84-derived centroids.

H3's within-band $\omega$ is dramatically smaller than either other grid's, at both resolutions and in both bands: a mean of $3.8°$–$3.9°$ and a ceiling of $13.14°$, against S2's mean of $15.7°$–$16.0°$ (ceiling $28°$–$30°$) and rHEALPix's mean of $22.0°$–$26.4°$ (ceiling $40°$–$46°$). This is the mirror image of Table 82.1's area-axis ranking, where H3 was the *least* area-uniform of the five grids measured ($2.328$–$2.367\times$, worse than S2's $1.59$–$1.89\times$ and far worse than rHEALPix's exact $1.000000$): on the angular axis, H3 is now the *most* uniform of the three grids with an $\omega$ computed here, by roughly a factor of 4–6 against S2 and 5–7 against rHEALPix. The maximum $\omega$ is identical to four significant figures across all four rows ($13.1404°$), consistent with it being reached at the same geometric configuration – the greatest great-circle distance from a face center attainable by an in-band cell centroid – in both bands and at both resolutions tested, rather than a coincidence of four separate computations. Widening the band from symmetric to asymmetric lowers H3's mean slightly ($3.91° \to 3.81°$ at res. 2) rather than raising it, the opposite direction from every area-axis figure that moved at all in Chapter 84: the newly included higher-latitude strips happen to sit, on average, slightly closer to their nearest face center than the original band's cells do.

**Key Result**

H3 closes out the three-grid angular-distortion comparison as the *most* angle-uniform DGGS measured in either band ($\bar{\omega} \approx 3.8°$–$3.9°$, ceiling $13.14°$), against S2 ($\bar{\omega} \approx 15.7°$–$16.0°$) and rHEALPix ($\bar{\omega} \approx 22.0°$–$26.4°$) – while being, on the area axis already reported in Tables 82.1 and 84.1, the *least* area-uniform of all five grids measured anywhere in this band-restricted comparison. Combined with Chapter 85's own S2-vs-rHEALPix finding, no grid measured in this band on both axes is uniform on both: S2 sits in the middle on area and the middle on angle; rHEALPix is exact on area and worst on angle; H3 is worst on area and best on angle. This is a genuine three-way trade, not a two-way one, and it sharpens rather than merely repeats Chapter 85's point that an equal-area construction's distortion does not vanish, it moves axis: H3's area-axis weakness is now shown to be closely tied to an angular-axis strength of comparable relative magnitude, on the same real cells, under the same band restriction.

**To Complete / Verify**

DGGRID's and OpenEAGGR's own within-band $\omega$ remain uncomputed; both are Snyder-ISEA (not per-face-gnomonic) constructions, so neither this chapter's closed-form solution

nor Chapter 31's Jacobian applies to them without first deriving or numerically differentiating their own map definitions, not attempted here. The traditional and optimized authalic/conformal spheres' own $\omega$ was separately closed already by Section 76.4.4, as Chapter 85's todobox itself now records; nothing here revisits that item. The band-placement question — whether any of this volume's area- or angle-based DGGS orderings hold at latitudes other than the two bands now tested — is unchanged by this chapter and remains the single item still open from Section 81.3's original four. Of that chapter's four-item chain, only the band-placement question is now left.
**Update:** Chapter 87 has since closed this last item, testing two new bands (equatorial and polar, each the symmetric band's own width) against all five DGGS. The area-axis ordering holds ordinally in both; the angle-axis ordering's "H3 most uniform" component holds in both, but its S2-vs-rHEALPix component reverses at the equator. Section 81.3's original four-item chain is now fully closed.

Chapter 87

# Does It Matter Where the Band Sits?

## 87.1 Scope: the last open item, finally tested

**Methodological Note**

Section 81.3's four-item chain closed three items across Chapters 82 and 85, and Chapter 86 closed a fourth, leaving exactly one: *the band-placement question — whether any of this volume's area- or angle-based DGGS orderings hold at latitudes other than the two bands now tested*. Every band chapter so far (Chapter 80 through Chapter 86) used one of exactly two bands, both derived from Kerkovits (2026)'s own $\Phi_0 = 45°$ construction: the symmetric band $[34.0988°, 55.2208°]$ and Chapter 77's wider, asymmetric $[27.3275°, 61.0330°]$. Both sit inside the same 27°–61° mid-latitude belt; neither says anything about the equatorial belt below it or the near-polar cap above it. This chapter tests two new bands chosen for exactly that reason, closing the volume's last open item from this four-chapter chain.

## 87.2 Method

The two new bands share the symmetric band's own angular width, $21.1220°$, so that cell counts stay of the same order as every earlier band chapter's, but are placed to fall entirely outside the already-tested asymmetric band's $[27.3275°, 61.0330°]$ footprint:

$$\text{equatorial} = [0.0000°, 21.1220°], \qquad \text{polar} = [68.8780°, 90.0000°] = [90° - 21.1220°, 90°].$$

These are arbitrary placements, not a re-derivation of Kerkovits (2026)'s own optimal band for some other $\Phi_0$; the width match and the non-overlap with the already-tested belt are the only design choices, and both are checked programmatically in `code/band_placement_sweep.py` before any grid is touched.

Every real-geometry engine used is the same one already built and verified elsewhere in this volume, generalized from a hard-coded band to an arbitrary $(\Phi_{\text{lo}}, \Phi_{\text{hi}})$ pair, with no change to its own internal method:

- **S2** (unit sphere): Chapter 80's exhaustive `CellId.begin(level)/.next()` sweep and `Cell.exact_area()`, levels 4 and 5, plus Chapter 85's finite-difference Tissot Jacobian (central difference, $\varepsilon = 10^{-4°}$) for $\omega$.
- **rHEALPix** (WGS84 for area, unit sphere for the Jacobian, the same split Chapters 80 and 85 already used): `RHEALPixDGGS.grid(r)`, resolutions 2 and 3.

- **H3** (`h3-py` v4): Chapter 81's area method and Chapter 86's closed-form per-face-gnomonic $\omega = 2\arcsin\left(\frac{h-k}{h+k}\right)$, $h = \sec^2 c$, $k = \sec c$, resolutions 2 and 3.
- **DGGRID** (ISEA7H): a fresh source build (DGGRID 9.0b, `github.com/sahrk/DGGRID`, commit `04bf5ed`, native CMake build) re-running `GENERATE_GRID`, `clip_subset_type WHOLE_EARTH`, resolutions 2/3/4, densification 0. The resulting whole-earth cell counts — 492, 3,432, 24,012 — match Chapter 82's own reported totals exactly, confirming this is the same global tessellation, only re-filtered by a different band. Area from the real AIGEN boundary polygon on the authalic sphere (pyproj `Geod`, $a = 6{,}371{,}007.2$m), exactly as that chapter computed it.
- **OpenEAGGR** (ISEA3H): a fresh source build (`riskaware-ltd/open-eaggr`, all 35 `Src/+API/` C++ files, statically linked against the repository's own bundled `libgdal.a/libproj.a` into a new `libeaggr.so`); `EAGGR_GetVersion` confirms `"v2.0"`, matching every other OpenEAGGR chapter in this volume. Enumeration, WKT outline, pyproj WGS84 polygon area, and mean-of-vertices centroid latitude are all identical to Chapter 82's own method, resolutions 2 and 4.

No $\omega$ is attempted for DGGRID or OpenEAGGR, for the same reason Chapter 86's own todobox already gives: both are Snyder equal-area (ISEA) constructions, not per-face gnomonic ones, and no closed-form or Jacobian machinery for either has been derived anywhere in this volume.

**Methodological Note**

**A genuine artifact, caught and excluded, not smoothed over.** At resolutions 2 and 3, rHEALPix's polar-cap cells N44 and S44 have centroids at exactly $90.0000°$ and $-90.0000°$ — the pole itself. The central-difference Tissot Jacobian needs $\Phi \pm \varepsilon$; at the pole, $\Phi + \varepsilon$ falls outside $[-90°, 90°]$ and longitude itself is undefined, so the computation is singular, not merely noisy: an early, unfiltered run of this chapter's own script returned $\omega = 180°$ for these two cells specifically, a chart artifact rather than a measurement. `code/band_placement_sweep.py` now detects $90° - |\Phi| < \varepsilon$ and excludes such cells from the $\omega$ statistics, reporting the exclusion count rather than silently dropping it; two cells are excluded at each of resolutions 2 and 3 in the polar band (none in the equatorial band, which does not reach the pole). Their real cell *areas* are unaffected — area involves no finite-difference step — and remain in the area-axis statistics.

## 87.3 Results: area axis

Table 87.1: Within-band normalized-area ratio (max/min), all five DGGS, in the two new bands. Source: `code/band_placement_sweep.py`, `data/band_placement_sweep_results.json`.

| grid | resolution(s) | equatorial $[0, 21.122°]$ | polar $[68.878°, 90°]$ |
|---|---|---|---|
| rHEALPix | res. 2, 3 | 1.000000 | 1.000000 |
| OpenEAGGR ISEA3H | res. 2, 4 | 1.0456, 1.0537 | 1.0090, 1.0291 |
| DGGRID ISEA7H | res. 2, 3, 4 | 1.1741, 1.3888, 1.1582 | 1.0137, 1.0452, 1.0135 |
| S2 | level 4, 5 | 1.7875, 1.9387 | 1.2996, 1.3884 |
| H3 | res. 2, 3 | 2.3280, 2.3673 | 1.6486, 1.7292 |

The five-grid ordering Chapter 82 established in the mid-latitude belt — rHEALPix (exact) $<$ DGGRID $\approx$ OpenEAGGR $<$ S2 $<$ H3, most to least area-uniform — **holds, ordinally, in both new bands**, without exception at either resolution tested for either DGGS. What does *not* hold is the magnitude: DGGRID's ratio at the equator ($1.39\times$ at res. 3) is nearly $30\times$ its own polar-band figure ($1.045\times$ at the same resolution) and more than triple its own already-reported mid-latitude figure ($1.046\times$, Table 82.1). It still stays below S2's equatorial minimum ($1.7875\times$), so the *ranking* survives even though this particular grid's own within-band uniformity is far more band-placement-sensitive than rHEALPix's, OpenEAGGR's, S2's, or H3's own figures, none of which move by more than roughly a factor of $1.5$ between the two new bands.

Section 82.3's unexplained non-monotonicity in DGGRID's resolution 2/3/4 sequence (res. 3 higher than both neighbours) reproduces in *both* new bands, at the same three resolutions, with the same qualitative shape ($1.174 \to 1.389 \to 1.158$ at the equator; $1.014 \to 1.045 \to 1.014$ at the pole). This rules out one of that todobox's three candidate explanations – "a coincidence of these three resolutions specifically" combined with the one band then tested – since the same shape now appears at three bands independently placed. It does not distinguish between the other two (genuine resolution-dependent structure in ISEA7H's hexagon/pentagon mix, versus a general property of aperture-7 subdivision unrelated to band placement); that remains open below.

## 87.4 Results: angle axis

Table 87.2: Within-band $\omega$ (degrees), mean, S2/rHEALPix/H3 (the three grids with a closed-form or Jacobian-based $\omega$ available anywhere in this volume). Source as Table 87.1.

| grid | resolution(s) | equatorial | polar |
|---|---|---|---|
| H3 | res. 2, 3 | $3.900°$, $3.891°$ | $3.076°$, $3.112°$ |
| S2 | level 4, 5 | $10.098°$, $10.157°$ | $7.079°$, $6.473°$ |
| rHEALPix | res. 2, 3 | $6.732°$, $6.638°$ | $22.738°$, $26.568°$ |

**Key Result**

The angle-axis ordering does *not* generalize the way the area axis does. At the polar band, Chapter 86's mid-latitude ordering H3 $<$ S2 $<$ rHEALPix (most to least angle-uniform) holds unchanged: $3.1° < 6.5$–$7.1° < 22.7$–$26.6°$. At the equatorial band it does not: S2 and rHEALPix trade places. rHEALPix's mean $\omega$ there ($6.6$–$6.7°$) is *lower* than S2's ($10.1$–$10.2°$), the opposite of every other band measured in this volume. H3 stays most angle-uniform in both new bands, by a wide margin ($3.1°$–$3.9°$ against either other grid's $6.5°$ or higher), so the one part of the ordering that survives everywhere tested is "H3 first"; the relative order of the other two is band-dependent, not fixed.

This is not an arithmetic fluke of one resolution: both S2 levels and both rHEALPix resolutions agree with each other within their own grid at each band, and the crossover is large relative to either grid's own resolution-to-resolution spread ($< 0.1°$ for S2, $< 0.2°$ for rHEALPix, against a $3.4$–$3.5°$ gap between the two grids at the equator). The qualitative shape is consistent with each grid's own already-documented distortion geometry elsewhere in this volume, though this chapter does not derive the mechanism further than pointing at it: rHEALPix's equal-area construction is already known (Chapter 101) to carry its angular cost unevenly across latitude, and S2's cube-to-sphere gnomonic projection is already known (Chapter 32) to carry its own worst distortion near cube-face edges and corners rather than uniformly; *where* each grid's

low-distortion regions happen to fall relative to the equator, specifically, is not derived here.

## 87.5 What this closes, and what it does not

**Key Result**

Of the two orderings this volume has built across five real DGGS, one generalizes past the single mid-latitude belt every earlier band chapter tested and one does not. The area-axis ranking (rHEALPix $<$ DGGRID $\approx$ OpenEAGGR $<$ S2 $<$ H3) holds ordinally in two new, non-overlapping bands spanning the equator to the pole, though individual grids' absolute figures — DGGRID's most of all — shift by up to an order of magnitude with band placement. The angle-axis ranking's "H3 most uniform" component also holds in both new bands; its S2-vs-rHEALPix component does not, reversing entirely at the equator. Section 81.3's original four-item chain is now fully closed: this is not a partial answer left for a future chapter, but the band-placement question specifically only ever asked whether the orderings hold elsewhere, not why – and the "why" of both the DGGRID non-monotonicity and the S2/rHEALPix angle-axis crossover remain open, honestly, below.

**To Complete / Verify**

Three things this chapter does not do, stated plainly rather than implied by omission. First, two additional bands are not a continuous sweep: the equatorial-to-polar crossover point where S2 and rHEALPix's angle-axis ordering flips is bracketed (between $21.122°$ and $68.878°$) but not located, and nothing here rules out a second crossover at some other latitude neither new band happens to sample. Chapter 88 closes this item: the crossover is located at four longitudes, and turns out not to be a single latitude at all — three crossings at longitude $0°$ alone, one each at the other three sampled longitudes. Second, DGGRID's non-monotonic resolution 2/3/4 sequence is now confirmed reproducible across three independently placed bands, which narrows its explanation to genuine ISEA7H structure rather than band-specific coincidence, but the chapter stops at narrowing, not deriving, that structure. Third, no $\omega$ figure exists anywhere in this volume for DGGRID or OpenEAGGR, in any band; closing that would require deriving or numerically differentiating each DGGS's own Snyder-ISEA map definition from scratch, not attempted here or anywhere earlier in this volume. Chapter 89 closes this item too: not by a from-scratch derivation (a first attempt at one did not reproduce DGGRID's real output and was abandoned rather than debugged), but by finite-differencing the actual compiled DGGRID binary's own output directly, extended to OpenEAGGR by a direct, term-by-term comparison of the two libraries' own source rather than by running OpenEAGGR itself.

Chapter 88

# Where the Angle-Axis Ordering Actually Flips

## 88.1 Scope: the first of three remaining items

Chapter 87's own todobox left three things undone; this chapter closes the first. That chapter's two new bands showed S2 and rHEALPix trading places on the angle axis between the equatorial band and the polar band, and stated plainly that "the equatorial-to-polar crossover point where S2 and rHEALPix's angle-axis ordering flips is bracketed (between $21.122°$ and $68.878°$) but not located, and nothing here rules out a second crossover at some other latitude neither new band happens to sample." Both halves of that caveat turn out to matter: the crossover is locatable, and there is, in fact, more than one of them at some longitudes.

**Methodological Note**

**What this chapter changes in method, and why.** Chapter 87's own angle-axis table used *band-mean* $\omega$ — one number per grid, per band, averaging over every cell's own longitude and resolution inside that band. That is adequate for asking whether an ordering holds across a band, but not for asking where, within a band, a crossing actually sits: a mean cannot be inverted to find a point. This chapter instead evaluates both grids' own point-wise Tissot $\omega$ — the identical closed-form/finite-difference machinery already verified in `code/kerkovits_vs_dggs_band_shape_metric.py` and `code/band_placement_sweep.py`, copied unchanged, not re-derived — as a continuous function of latitude at a fixed longitude, and finds the sign change of $\omega_{S2}(\Phi) - \omega_{\text{rHEALPix}}(\Phi)$ by bisection. Full method and code: `code/s2_rhealpix_angle_crossover.py`, `data/s2_rhealpix_angle_crossover_results.json`.

## 88.2 Method

Both grids have longitude-dependent distortion fields — S2 from its six cube faces meeting the equator in a pattern with a documented period (Chapter 32), rHEALPix from its four lateral quadrant faces plus polar caps (Chapter 101) — so a single longitude slice would not be representative. Four longitudes are used: $0°$, $22.5°$, $45°$, $67.5°$, spanning one full quadrant ($90°$) of both grids' own approximate four-fold symmetry in longitude, at $22.5°$ spacing. This spacing is not derived from either grid's actual face boundaries, which differ between the two grids and are not aligned to each other; it is a deliberately naive, symmetry-agnostic sampling, stated as such before any result is shown, not chosen after the fact to avoid a known problem. Section 88.4 shows this naivety was not free.

At each longitude, $\mathrm{diff}(\Phi) = \omega_{S2}(\Phi) - \omega_{\mathrm{rHEALPix}}(\Phi)$ is evaluated on a coarse $1^\circ$ grid from $1^\circ$ to $89^\circ$, every sign change is located (more than one is not assumed impossible *a priori*, and, as Section 88.3 shows, this caution was warranted), and each is refined by bisection to $10^{-4\circ}$. A finer $0.25^\circ$ coarse grid was re-run at every longitude as a check; it finds exactly the same sign changes, so the $1^\circ$ grid was not hiding anything nearby.

**Methodological Note**

**A second artifact, caught the same way Chapter 87's polar-cap one was.** Every candidate sign change is probed at 20 points across its own bracket before being accepted; a genuine crossing shows a smooth approach to zero, while a chart seam — a discontinuity in the underlying map, not a real change in physical distortion — shows a jump. One such seam is caught this way, at longitude $0^\circ$, bracketed in $[41^\circ, 42^\circ]$: this is rHEALPix's own ordinary face-boundary transition at that longitude, not a crossing, and is excluded from the results below exactly as Chapter 87 excluded its own pole-singularity artifact — reported, not silently dropped.

## 88.3 Results: the crossover is not one point

Table 88.1: S2/rHEALPix angle-axis crossing latitudes, by longitude. Residual is $\mathrm{diff}(\Phi)$ at the bisected latitude (should be $\approx 0$; largest magnitude shown is $5.7 \times 10^{-5}$ degrees of $\omega$, four orders of magnitude below the smallest gap between crossings). Source: `code/s2_rhealpix_angle_crossover.py`, `data/s2_rhealpix_angle_crossover_results.json`.

| longitude | crossing lat. (°) | $\omega$ at crossing (°) | note |
|---|---|---|---|
| $0^\circ$ | 6.9822 | 8.5300 | |
| $0^\circ$ | 39.7813 | 20.6709 | |
| $0^\circ$ | 56.5597 | 20.8083 | |
| $0^\circ$ | — | — | seam artifact in $[41^\circ, 42^\circ]$, excluded |
| $22.5^\circ$ | 35.2109 | 13.7321 | |
| $45.0^\circ$ | 41.4467 | 23.4747 | contaminated below $35.2644^\circ$; see Section 88.4 |
| $67.5^\circ$ | 35.2109 | 13.7321 | identical to $22.5^\circ$ to 4 decimals |

**Key Result**

Chapter 87's todobox asked for one thing and its caveat anticipated the possibility of more; both parts of that were right to be asked, and the answer is more than one. At longitude $0^\circ$ alone, the S2/rHEALPix angle-axis ordering flips *three* times between the equator and the pole, not once: at $6.98^\circ$, $39.78^\circ$, and $56.56^\circ$. The other three sampled longitudes each show exactly one crossing in this range, at $35.21^\circ$ ($22.5^\circ$ and $67.5^\circ$, agreeing to four decimal places) and $41.45^\circ$ ($45^\circ$, subject to the caveat in Section 88.4). There is therefore no single "the" crossover latitude: the point where the ordering flips is itself longitude-dependent, consistent with the docstring's own starting premise that neither grid's distortion field is longitude-independent, and Chapter 87's two-band sweep, sampling only $\Phi = 0^\circ$–$21.122^\circ$ (mean, all longitudes pooled) and $\Phi = 68.878^\circ$–$90^\circ$, could not have distinguished "one crossing" from "three" because it never asked a longitude-resolved question in the first place.

The $22.5°/67.5°$ agreement is not a coincidence requiring further explanation: those two longitudes are mirror images of each other about $45°$ within the shared $90°$ period both grids' own four-fold symmetry already implies ($67.5° = 90° - 22.5°$), so identical crossing latitudes are exactly what that symmetry predicts, and their agreement to four decimal places is a consistency check on the sweep's own correctness rather than a new finding in its own right.

## 88.4 A longitude that lands exactly on a cube edge

**Methodological Note**

**What was found, how it was checked, and what it does not change.** The naive $22.5°$-spaced longitude sample places one sample, $45°$, exactly on an S2 cube-face boundary for a wide latitude range, not merely near one. Along the meridian $\lambda = 45°$, $x = \cos\Phi\cos 45°$ and $y = \cos\Phi\sin 45°$ are equal at every latitude ($\cos 45° = \sin 45°$), so S2's own largest-component face selection is in a live tie between face 0 and face 1 along the whole meridian, not just at one point. In exact arithmetic this tie breaks only where $|z|$ overtakes $|x| = |y|$, at the cube-vertex latitude $\arcsin(1/\sqrt{3}) = 35.2644°$; `code/s2_lon45_edge_artifact_check.py` confirms this in two independent ways: (i) direct evaluation matches the closed-form prediction to $10^{-4}$ degrees, and (ii) floating-point rounding of the live tie produces a narrow band of erratic face-0/face-1 flicker below that latitude (a numerical curiosity of the tie itself, distinct from the substantive finding), after which face 2 is selected stably and permanently.

Below $35.2644°$, the point sits *on* the shared face-0/face-1 edge, where the raw projective coordinate $u$ is pinned at exactly $1.0$ rather than varying smoothly with longitude; the central-difference longitude derivative $\partial(S,T)/\partial\lambda$ that the Tissot Jacobian needs straddles this edge and is degenerate there. The result is not a smoothly large but physically ordinary $\omega$: $\omega_{S2}$ is inflated to $179.6$–$179.7°$ throughout this band (checked at seven latitudes from $1°$ to $35°$), a chart artifact of the kind Chapter 32 already documents for a different transect (the diagonal $\lambda = \phi = \theta$, approaching the same cube vertex from a different direction), not a real feature of either grid's distortion. This is a genuinely new instance of that chapter's own todobox item, left explicitly open there: "[the earlier chapter] does not establish where the face-0/face-2 boundary falls in general (away from the diagonal)." It still does not establish this in general; it establishes one further case, a constant-longitude transect landing exactly on a two-face edge rather than crossing one transversally, which is a different failure mode from the diagonal transect's corner-adjacent one.

**Key Result**

The single crossing this sweep reports at longitude $45°$ ($\Phi = 41.4467°$) is not itself contaminated: it lies $6.18°$ past the cube vertex, and the smoothness probe — the same 20-point-per-bracket check used everywhere else in this chapter — confirms the interval $[36°, 50°]$ containing it is smooth on both sides, while $[30°, 36°]$, which spans the transition, is not. The reported *crossing latitude* at $45°$ longitude therefore stands; what does *not* stand is any equatorial-band $\omega$-difference figure at that longitude, since every latitude below $35.2644°$ there is chart-artifact-dominated (diff $\approx$ 166–170°, not a real angle-uniformity gap between the two grids). No other sampled longitude sits on a face edge this way, checked directly in Section 88.3's own table generation.

## 88.5 What this closes, and what it still does not

Chapter 87's first todobox item asked for the crossover to be located rather than merely bracketed. It now is, at four longitudes, each to $10^{-4}$ degrees by direct bisection on the point-wise Tissot machinery already verified elsewhere in this volume — but the located answer is not the single number the question's own phrasing presupposed. The crossover is longitude-dependent, and at longitude $0°$ specifically there are three of them, not one, inside the same range Chapter 87's two bands bracket. This closes the item honestly rather than by supplying the oversimplified single figure the original phrasing invited.

**To Complete / Verify**

Four longitudes at $22.5°$ spacing are not a continuous sweep in longitude, and this chapter does not claim they are: nothing here rules out additional crossings at some other longitude none of the four samples, nor establishes how many crossings exist in general as a function of longitude (three at $0°$, one at the other three sampled points — the pattern connecting these, if any, is not derived). Second, the face-edge/tie mechanism found at $45°$ longitude is characterized only along that one meridian; whether analogous live ties recur at other longitudes for either grid, and under what condition, is not derived in general. Third, Chapter 87's other two open items — DGGRID's non-monotonic resolution sequence, and the absence of any $\omega$ figure for DGGRID or OpenEAGGR — remain exactly as open as that chapter left them; nothing here touches either.

Chapter 89

# Closing the Gap: The Indicatrix, Real and Measured, for DGGRID and OpenEAGGR

## 89.1 Scope: the gap left open twice

Two earlier chapters left the same gap open from two different directions. Chapter 36 derived a real Tissot indicatrix for H3's gnomonic projection in closed form, then refused to attempt DGGRID or OpenEAGGR, since "DGGRID's ISEA7H grid is a different, independently-implemented realization of the same conceptual (icosahedral, aperture-7) construction" and OpenEAGGR's ISEA3H a different aperture again, neither re-derived there. Section 87.5's todobox, much later in the volume, restated the same gap from the band-placement side: "no $\omega$ figure exists anywhere in this volume for DGGRID or OpenEAGGR, in any band." This chapter closes it – but not by the route a first attempt took.

**Methodological Note**

**What was tried, what failed, and what was done instead.** A from-scratch Python port of DGGRID's own `DgProjISEA.cpp` (the sphere-to-plane radial-azimuthal law, function `sllxy`) and `DgProjTriRF.cpp` (the twenty-triangle icosahedron construction and face-selection logic, `ico12verts/ whichIcosaTri`) was attempted first. Checked against DGGRID v9.0b (commit `04bf5ed`, the same commit already cited in this volume's bibliography via Sahr (2026)) built from source and run with `TRANSFORM_POINTS/ PROJTRI` on ten real test points, the port did not reproduce DGGRID's real output: every one of the ten points disagreed, several by a wide margin, and none matched to the tolerance the port's own author intended. Rather than debug that port's icosahedron-construction code – a software-engineering exercise with no cartographic content of its own – this chapter instead queries the real compiled DGGRID binary directly as a black box: a strictly stronger no-invented-data guarantee than even a correct port would have offered, and the same standing this volume already gives DGGRID elsewhere (Chapter 15). Two genuine limitations of DGGRID's own `PROJTRI` text output surfaced in doing this, both found by direct probing and both worked around rather than hidden: first, the output is hard-capped at six decimal digits (traced to `DgRFBase::setFormatStr()`, whose printf-style format string is built from a precision left at the class's constructor default of 6 for this particular reference frame – the `precision` meta-parameter is applied only to grid-statistics output elsewhere in the source, never here), which floors the usable finite-difference step size; second, at low `dggs_res_spec` the coordinates are snapped to a resolution-linked quantization grid (order 0.01–0.08 in the $[0, 1]$ face frame at resolution 4), which a finite-difference step smaller than the grid spacing cannot see through at all. Resolution 15 was found (by comparing resolutions 12/15/18, which agree to six decimals) to be fine enough that this quantization no longer limits anything, and is used throughout. Full code:

`code/tissot_dggrid_real.py`; full output: `data/tissot_dggrid_real_results.json`.

## 89.2 Locating a real face center, and a real transect

DGGRID's own `PROJTRI` coordinate convention places every triangle's center at the same fixed point in that triangle's own normalized frame: $(x, y) = (0.5,\ 1/(2\sqrt{3}))$, since `DgProjISEA.cpp`'s own constants give `originXOff/icosaEdge`$= 0.5$ exactly (`icosaEdge`$= 2\times$`originXOff`) and `originYOff/icosaEdge`$= 0.34773.../1.20459...$, the standard equilateral-triangle centroid offset. Rather than transcribe a vertex table, the real geographic point mapping to that fixed target was located by Newton iteration directly against the compiled binary (each step's Jacobian obtained from real finite differences of real `PROJTRI` output), converging in four iterations to triangle 2's true center at $11.250017°$E, $20.905123°$N – itself a small independent check, since the longitude lands within $0.00002°$ of DGGRID's own default vertex-0 longitude ($11.25°$, Section 36.1's H3 chapter cites the analogous H3 face-center table by contrast), consistent with the icosahedron's own five-fold symmetry about that vertex rather than independently re-derived here.

From that real center, two radial transects were run outward by real spherical bearing (bearing $0°$, due north, and bearing $90°$, due east), out to $35°$ – short of the triangle's own circumradius, $37.377°$ (= the constant `DH` in `DgProjISEA.cpp`, and, not by coincidence, the same icosahedron face circumradius Chapter 36 computed independently from H3's own face-center table: both constructions tile the same regular icosahedron, only the face-internal projection differs).

Table 89.1: Real DGGRID ISEA7H Tissot indicatrix, triangle 2, two radial transects from the real (Newton-located) face center. $\omega$ and area scale from finite-differencing the actual compiled binary's own `PROJTRI` output (step $0.02°$). Area scale is reported in DGGRID's own normalized `PROJTRI` units (divided by `icosaEdge`), not true sphere-radius units – see Section 89.3. Source: `code/tissot_dggrid_real.py`, `data/tissot_dggrid_real_results.json`.

| $z$ (°) | $\omega$, bearing 0° (°) | $\omega$, bearing 90° (°) | area scale (both bearings) |
|---|---|---|---|
| 0 | 1.685 | 1.685 | 0.698 |
| 5 | 9.202 | 7.135 | 0.689 |
| 10 | 8.963 | 7.247 | 0.690 |
| 15 | 8.367 | 7.401 | 0.688 |
| 20 | 7.555 | 7.687 | 0.689 |
| 25 | 6.561 | 8.065 | 0.690 |
| 30 | 5.386 | 9.989 | 0.688 |
| 35 | 3.937 | 15.541 | 0.688 |

*bearing-90° points at $z \geq 25°$ fall in the neighboring triangle 3, not triangle 2; both real triangles' figures are shown together since the transect crosses a real internal edge, not a computational artifact (each side's own `PROJTRI` triangle index confirms this).

## 89.3 An equal-area check that falls out of the data itself

**Key Result**

The area-scale column above is constant to within $\pm 0.3\%$ relative across sixteen points spanning two different bearings, two different triangles, and the full $0°$–$35°$ radial range – exactly the signature an equal-area projection must show (constant area scale, freely varying angular deformation), and not assumed here but read directly off the measured data. Its actual numerical value, $\approx 0.689$, is not $1$ only because DGGRID's own `PROJTRI` coordinates are normalized by `icosaEdge` rather than reported in true sphere-radius units: $(1/\texttt{icosaEdge})^2 = (1/1.2045910058)^2 = 0.6891$, matching the measured constant to three significant figures. This is an independent numerical confirmation of DGGRID's own coordinate convention, obtained without trusting the earlier (and, as Section 89.1 disclosed, unreliable) from-scratch port at all.

The $\omega$ values themselves are not small: single digits near the face center, rising past $9°$ within $5°$ of it along one bearing, and reaching $15.5°$ near the far edge along the other. These real, measured figures for the actual (untruncated, 20-triangle) icosahedron exceed Snyder (1992)'s own abstract-reported bound of $3.75°$ angular deformation – a bound Chapter 19 already flagged as belonging to a *truncated* icosahedron and explicitly not known to transfer to the plain 20-triangle construction ISEA-family DGGS actually use. This chapter's real figures confirm that caution was warranted: the bound does not transfer, at least not at this face and this radial range.

## 89.4 Step-size stability, and the honest precision ceiling

Table 89.2: Step-size convergence, $z = 20°$, bearing $0°$. Source: `data/tissot_dggrid_real_results.json`.

| $\varepsilon$ (°) | $\omega$ (°) |
|---|---|
| 0.1 | 7.592 |
| 0.05 | 7.611 |
| 0.02 | 7.555 |
| 0.01 | 7.648 |
| 0.005 | 7.944 |

The figure is stable to within about $\pm 0.1°$ across a factor-20 range of step sizes ($0.005°$–$0.1°$), degrading only at the smallest step tested – the expected signature of Section 89.1's disclosed six-decimal output ceiling starting to bind, not of the underlying map becoming unstable. This is a real, disclosed precision floor around a few hundredths of a degree in $\omega$, not the nine-decimal-place agreement Chapter 36 obtained for H3's closed-form gnomonic map (whose source exposes full double precision directly): DGGRID's own text output, not this chapter's method, is the limiting factor, and the figures above are reported to the precision that ceiling actually supports.

## 89.5 OpenEAGGR: the same map, confirmed from its own source

OpenEAGGR's forward projection (RiskAware (2017), files `Snyder.cpp` and `Icosahedron.cpp` under `EAGGR/Src/Model/IProjection` and `EAGGR/Src/Model/IPolyhedralGlobe` respectively, read directly rather than run) was compared term for term against DGGRID's `DgProjISEA.cpp`. The two implement the same sequence of equations – spherical distance and azimuth from a face center, an azimuth adjustment folding the angle into a 120° sector, the same radial law built from $q$/`dz`, $H$/`h`, $AG$/ `ag`, $Az'$/`azh1`, $d'$/`fh`, and finally $\rho$/ `ph` – under different variable names but identical operations in identical order, with identical numerical constants: OpenEAGGR's `Get_g()` ($37.37736814°$), `Get_G()` ($36.0°$), `GetTheta()` ($30.0°$), and `GetRPrimeRelativeToR()` ($0.91038328153090290025$) match DGGRID's `DH`, `GH`, the 30° implicit in `cot30`, and `R1` to nine significant figures or better. Only the twenty face centers' fixed placement on the sphere differs between the two libraries (OpenEAGGR's own `GetFaceCentre()` table, degrees $E = 52.62263186$, $F = 10.81231696$, versus DGGRID's vertex-0-based construction) – a purely position-independent choice of where each face sits on the globe, exactly analogous to the fixed rotations Chapter 36 already argued cannot alter $\omega$ at a given point *within* a face. On the strength of that direct, term-by-term source identity – not a numerical run of OpenEAGGR's own compiled library, which this chapter does not attempt – the $\omega(z, \text{bearing})$ field measured in Table 89.1 for DGGRID's real output applies equally to OpenEAGGR's ISEA3H: the same shared Snyder ISEA map underlies both, placed differently on the sphere but not differently shaped on any one face.

**To Complete / Verify**

This chapter closes the gap for one face (triangle 2) of DGGRID's real ISEA7H, measured from the actual compiled binary rather than a port, and extends the result to OpenEAGGR by direct, term-by-term source comparison rather than by running OpenEAGGR's own library. Three things it does not do. First, only one face was measured; the icosahedron's own symmetry makes the other nineteen equivalent up to a rigid rotation, but that equivalence is asserted from the construction's known symmetry, not independently re-verified face by face here. Second, the face-center value itself ($\omega \approx 1.7°$ at $z = 0°$, Table 89.1) is small but not exactly zero; the residual is attributed to the combination of the Newton search's own convergence tolerance and finite-difference step size near a point where all partial derivatives are themselves small, in line with Section 89.4's disclosed precision floor, but this is not separately isolated from a possible true (if small) analytic non-vanishing at the center. Third, bearing 0°'s non-monotonic profile in Table 89.1 – rising sharply from the center, then falling most of the way back down by $z = 35°$ – is reported as measured, not mechanistically explained: unlike Chapter 36's closed-form gnomonic case, this map's piecewise, 120°-sector-folded construction has no closed form derived anywhere in this volume, so nothing here identifies which part of the construction produces that shape.

Chapter 90

# New Zealand in the Band: Connecting the National Case to the Abstract Comparison

## 90.1 Scope: the connection Chapter 79 left for later

**Methodological Note**

**What this chapter does and does not do.** Chapter 79's closing todobox left its own numbers unrelated to Chapter 80's and Chapter 78's, deferring that link "once this volume's existing regional chapters are back in view together with this one." This chapter makes exactly that link, and no new computation: every number below is already published, in Table 79.1 or Table 80.1. What is new is the observation that connects them, and it is a genuine one, not a foregone conclusion – it was not designed into either chapter, since Chapter 79 was written independently of Chapter 80 (Section 79.1).

## 90.2 The coincidence: one country, one matched band

Chapter 80's within-band area-ratio comparison (Table 80.1) is restricted to $[34.0988°, 55.2208°]$ – the acceptance interval Kerkovits (2026)'s own rectifying sphere derivation produces for $\Phi_0 = 45°$, reproduced independently in Section 76.4.1. Chapter 79's seven New Zealand locations span $34.4$°S (Cape Reinga) to $46.6$°S (Bluff), with the Chatham Islands at $44.0$°S. All seven fall inside that same band – not approximately, but entirely: even Cape Reinga, the northernmost point and the one closest to the band's lower edge, sits $0.3°$ inside it, not on it. New Zealand was not chosen for this reason (Section 79.1 gives its own, unrelated justification, taken from LINZ's own documentation), and Chapter 80's band was fixed by $\Phi_0 = 45°$ alone, with no reference to any specific country. The overlap is therefore a genuine coincidence of two independently motivated choices, not a constructed match – but it means Chapter 80's abstract band comparison and Chapter 79's concrete national case describe, without either chapter intending it, the same stretch of latitude.

Table 90.1: The same latitude band, two chapters' figures side by side. NZ $\omega$ range from Table 79.1 (all seven points, all inside the band per Section 90.2); band area ratio from Table 80.1.

| Map | $\omega$ range, in band (Chapter 79) | Area ratio, in band (Chapter 80) |
|---|---|---|
| S2 | $11.146°$–$15.575°$ | 1.593786–1.761795 |
| rHEALPix | $12.653°$–$46.423°$ | 1.000000 |
| NZTM2000 | $\approx 0°$ (conformal) | not computed by either chapter |

## 90.3 What lines up, once the band is shared

**Key Result**

Chapter 80's integritybox (Section 80.2) warned against summing or ranking area-uniformity and angular-uniformity as though they were one distortion score. New Zealand's real geography now supplies a concrete instance of exactly why: within this single $21°$ band, rHEALPix is area-perfect (1.000000, by construction) and simultaneously the most angularly distorted of the three maps this volume has computed for these seven points ($\omega$ up to $46.423°$, more than three times its own best value). S2, over the identical band, shows the reverse pairing – real, non-trivial area non-uniformity ($1.59$–$1.76\times$) alongside comparatively restrained angular distortion ($11.1°$–$15.6°$, a narrower range than either of the other two grids' most distorted figures). No new method was needed to see this: it was already implied by Table 79.1 and Table 80.1 separately, and only becomes visible once, as here, both are read against the same stretch of latitude.

**Key Result**

Chapter 78's opening question – what a region buys and gives up by building a low-distortion surface around itself – now has a third answer sitting beside its abstract one. Chapter 79 already showed what NZTM2000 buys inside this exact band: angular distortion at the numerical floor, against $11$–$46°$ for the two global grids at the same latitudes. This chapter adds that the band itself is not an arbitrary yardstick invented for New Zealand – it is the same interval Chapter 80 already used to characterize those two global grids' area behaviour in the abstract, so the "price of the regional sphere" Section 79.4 describes in NZTM2000's case and the "price of the global grid" Chapter 80 describes in S2's and rHEALPix's are, for the first time in this volume, prices quoted over identical latitude.

**To Complete / Verify**

This chapter links two tables that already existed; it does not compute anything new, and in particular does not attempt Chapter 79's own still-open item – a continuous distortion field over New Zealand's full extent, or a cell-by-cell DGGS tessellation overlaid on the country, rather than seven point locations. It also does not compute an in-band area ratio for NZTM2000 itself, so the table above's third row is left explicitly blank rather than filled by a number neither source chapter produced.

Chapter 91

# A Second Documented National Case: Switzerland's Century-Old Local Frame

## 91.1 Scope: a second real data point, not a second method

**Methodological Note**

**Independence declared up front.** Like Chapter 79, this chapter is a self-contained addition to Part II, computed from scratch with the libraries already used elsewhere in this volume (`s2sphere`, `rhealpixdggs`, `pyproj`). It reuses Chapter 79's exact method — the same `tissot_from_jacobian()` construction, the same finite-difference step, the same step-size stability check — applied to a different country, so that Part II's single-case NZTM2000 finding rests on a second, independently chosen real dataset rather than one country's own possibly idiosyncratic geometry.

Switzerland is a documented instance of Part II's question that predates New Zealand's by nearly a century. Swisstopo (2026b) record that M. Rosenmund developed, in 1903, an oblique conformal cylindrical projection (a Swiss oblique Mercator) fixed to the old Bern observatory as its origin point, adopted as the official Swiss national reference system CH1903. Swisstopo (2026a) report that this reference system was modernized in two further steps: CHTRS95, tying the Swiss frame to the global GRS80/ETRS89 system as of 1993, and CH1903+/LV95, the current official system for national and cadastral surveying, moving the fundamental point to the Zimmerwald observatory and re-deriving the same oblique-Mercator construction on the updated geodetic base. Unlike New Zealand's NZMG-to-NZTM2000 transition (Chapter 79), which replaced one projection family (a bespoke complex-polynomial construction) with a geometrically standard one (transverse Mercator), Switzerland's transition kept the same oblique-Mercator projection formula across all three reference-system generations, updating only the underlying geodetic datum and fundamental point – a different history, but one that poses the same Part II question: what does a small, high-latitude region actually buy by using a projection built around itself, instead of a global DGGS face?

## 91.2 Method

**Methodological Note**

**What was computed.** Seven real locations were fixed by ordinary geographic coordinates: Geneva, Basel, Bern (at CH1903+/LV95's own projection centre), Zurich, Lugano, St. Moritz, and Vaduz in Liechtenstein (included because Liechtenstein also uses the Swiss CH1903+/LV95 system, per Swisstopo, 2026a, giving one point outside Switzerland's own borders but inside the reference system's actual domain of use). At each

point this chapter computed the Tissot indicatrix – $a$, $b$, $\omega$, and the area scale factor – for three maps, using Chapter 31's `tissot_from_jacobian()` (reused unchanged from `code/tissot_s2_rhealpix.py`), exactly as Chapter 79 did for NZTM2000:

- S2's own composite gnomonic-then-quadratic forward map (`s2sphere`), exactly as in Section 31.4;
- rHEALPix's own forward map (`rhealpixdggs`), exactly as in Section 31.3;
- CH1903+/LV95 (EPSG:2056), read from `pyproj`/PROJ's own authoritative definition rather than retyped from a document: Hotine Oblique Mercator variant B, projection centre $46.9524°$N/$7.4396°$E, azimuth $90°$, false easting $2\,600\,000$ m, false northing $1\,200\,000$ m, on the Bessel 1841 ellipsoid (EPSG/IOGP, 2026).

As in Chapter 79, CH1903+/LV95's finite-difference Jacobian (in metres per radian) is converted to a dimensionless point scale factor using the local meridional radius of curvature $M$ and prime-vertical radius of curvature $N$ at each point's latitude, since it is defined on a real ellipsoid rather than the unit sphere used for S2 and rHEALPix. Step-size stability was checked at Bern across $\varepsilon \in \{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ degrees; $\omega$ stayed at or below $1.3 \times 10^{-6\circ}$ at every step size tested, i.e. zero within the precision of the method. Full code: `code/tissot_swiss_regional.py`; full output: `data/tissot_swiss_regional_results.json`, `data/tissot_swiss_regional_convergence.json`.

## 91.3 Results: the tightest distortion budget in this volume, against two loose global ones

Table 91.1: Tissot indicatrix at seven Swiss/Liechtenstein locations, three maps. $\omega$ in degrees; the CH1903+/LV95 "$k$" column is the dimensionless point scale factor $k = \sqrt{\text{area scale}}$, the quantity used throughout the Swiss geodetic literature for this projection. Source: `code/tissot_swiss_regional.py`, `data/tissot_swiss_regional_results.json`.

| Location | S2 $\omega$ | rHEALPix $\omega$ | CH1903+ $\omega$ | CH1903+ $k$ |
|---|---|---|---|---|
| Geneva | 13.388 | 44.191 | $\approx 0$ | 1.000083 |
| Basel | 12.852 | 42.961 | $\approx 0$ | 1.000056 |
| Bern (centre) | 12.756 | 43.114 | $\approx 0$ | 1.000000 |
| Zurich | 12.278 | 42.217 | $\approx 0$ | 1.000028 |
| Lugano | 11.734 | 42.005 | $\approx 0$ | 1.000134 |
| St. Moritz | 11.534 | 41.270 | $\approx 0$ | 1.000029 |
| Vaduz (Liechtenstein) | 11.788 | 41.461 | $\approx 0$ | 1.000007 |

**Key Result**

CH1903+/LV95's angular distortion is zero to the precision of the method at all seven points, for the same structural reason Chapter 79 found for NZTM2000: an oblique Mercator projection is conformal by construction (Snyder, 1987), so $\omega \equiv 0$ is a property of the formula, not a discovery about Switzerland. What *is* a finding, because it required the computation rather than the conformal property alone, is how tight the accompanying area budget is: $k$ ranges only from $1.000000$ at the projection centre to $1.000134$ at Lugano, the furthest of the seven points from Bern — a maximum area distortion of about $0.013\%$,

an order of magnitude tighter than NZTM2000's own $k$ range of roughly $0.9996$ to $1.0083$ found in Chapter 79 across New Zealand's similarly-sized but geographically more elongated territory. Both projections are conformal by the same formula family (a Mercator variant); Switzerland's far tighter budget follows directly from its far smaller extent from the projection's own centre, not from any difference in projection design between the two countries.

## 91.4 The same global-scheme contrast, and one difference worth noting

At the same seven points, S2 shows $\omega$ between $11.5°$ and $13.4°$ – a narrower range, and at lower absolute values, than its own $11.1°$–$15.6°$ spread across New Zealand's seven points in Chapter 79 – while rHEALPix shows $\omega$ between $41.3°$ and $44.2°$, consistently in the upper part of the $12.7°$–$46.4°$ range Chapter 79 found for New Zealand, and, unlike that chapter's results, essentially flat across all seven Swiss points.

**To Complete / Verify**

The rHEALPix area-scale figure returned by this chapter's computation is identical to four decimal places $(1.1781)$ at all seven Swiss/Liechtenstein points, despite their spanning roughly $9°$ of longitude and $1.6°$ of latitude. This is consistent with, and offered here only as a plausible reading of, rHEALPix's own equal-area design (Gibb et al., 2016; Chapter 19): if the scheme is exactly equal-area, a constant area-scale factor across a small region is expected, and the varying $\omega$ values in Table 91.1 reflect changing local shape distortion at fixed area, not changing area distortion. This chapter does not independently re-derive rHEALPix's equal-area property from the computation above; it notes the observation and its most natural explanation, without treating the explanation as separately verified here.

## 91.5 Closing: two countries, one Mercator variant, two different margins

Read beside Chapter 79, this chapter's Swiss case strengthens Part II's single-country finding into a two-country pattern using a genuinely independent computation and a genuinely independent choice of country, projection generation, and data source: a small region built around its own conformal projection buys angular distortion indistinguishable from zero and an area budget tight enough to be a rounding error, at global schemes' cost of tens of degrees of angular distortion at the same latitudes. What differs between the two cases is the *size* of that regional advantage, not its existence: Switzerland's smaller extent from its own projection centre buys an area budget an order of magnitude tighter than New Zealand's, while both countries pay, in exchange, the same loss Chapter 79 already named — a projection understood and directly usable only inside the region it was built for, exactly the trade-off LINZ's own documentation stated in its own words for New Zealand, and exactly the trade-off implicit in Rosenmund's 1903 choice to fix a wholly new projection to a single observatory in Bern rather than adopt an already-existing global scheme.

# Part III

# The Cost of Distortion: Efficiency Against Accuracy

Chapter 92

# The Equal-Area / Computational-Efficiency Trade-off

## 92.1 The tension, stated by its own literature

Wecker et al. (2024) state explicitly the tension at the centre of this third part: equal-area DGGS are desirable for their low distortion, but fall short of that promise because equal-area projections are themselves computationally inefficient, while efficiency-first DGGS must address distortion less well (Wecker et al., 2024). The authors propose a mesh-based DGGS (MBD) that generalizes efficient operations on watertight triangular meshes with spherical topology, allowing high-resolution polyhedra to be used as the base solid in order to reduce distortion (Wecker et al., 2024).

**Methodological Note**

Wecker et al. (2024)'s record is fully verified: *ISPRS International Journal of Geo-Information*, volume 13, issue 11, article 373, DOI 10.3390/ijgi13110373, submitted 6 September 2024, revised 15 October 2024, accepted 19 October 2024, published 22 October 2024, authored by Lakin Wecker, John Hall, and Faramarz F. Samavati (Department of Computer Science, University of Calgary), confirmed against the publisher's own article page.

This is exactly the thread stated in this volume's preface: the same question — how much does it cost to represent something continuous discretely — now posed as an explicit trade-off between two costs rather than as a single distortion measure. This chapter takes the first step toward formalizing that trade-off as a constrained optimization problem, using quantities already derived and verified in Chapters 1 and 58 of this volume.

## 92.2 A toy constrained-optimization formalization

**Methodological Note**

Everything in this section is an explicitly labelled *toy* model, built entirely from the exact discrete-graticule results of Chapters 1 and 58. It illustrates the *qualitative* shape of the tension Wecker et al. (2024) describe in words; it is not a reproduction of any numeric result from their paper, and no claim below should be read as a statement about MBD, H3, S2, rHEALPix, DGGRID, or OpenEAGGR specifically. Unlike an earlier draft of this chapter, the toy model below is no longer this volume's only grounding for the trade-off: Theorem 59.6 (Chapter 59) proves, in general and without any toy simplification, that equal angular width and equal cell area cannot coexist for $K \geq 2$. What follows should be read as one specific, illustrative quantification of a trade-off already known to be forced, not as the argument that a trade-off exists at all.

**Definition 92.1** (Toy distortion and efficiency costs)**.** For the $K$-ring discrete graticule of Definition 1.1, define the *distortion cost*

$$D(K) \;:=\; \sup_i \bigl|\rho_i - \cos\phi_i\bigr|$$

(the uniform bulk error of Proposition 58.3), and the *efficiency cost*

$$E(K) \;:=\; K,$$

taken as a stylized proxy for indexing depth: a $K$-ring graticule requires on the order of $K$ distinguishable latitude bands to be addressed by any indexing scheme built directly on this tessellation.

**Proposition 92.2** (A toy Pareto relation)**.** *For the costs of Definition 92.1,*

$$D(K)\cdot E(K)^2 \;\longrightarrow\; \frac{\pi^2}{32} \qquad \text{as } K\to\infty. \tag{92.1}$$

*Proof.* By Proposition 58.3, $D(K) = \sec(\Delta\phi/2) - 1$ with $\Delta\phi = \pi/(2K)$, and $D(K) = (\Delta\phi)^2/8 + O((\Delta\phi)^4) = \dfrac{\pi^2}{32K^2} + O(K^{-4})$. Multiplying by $E(K)^2 = K^2$ gives $D(K)E(K)^2 = \dfrac{\pi^2}{32} + O(K^{-2}) \to \dfrac{\pi^2}{32}$. □

Table 92.1 verifies Eq. (92.1) numerically: the product $D(K)\cdot E(K)^2$ converges to $\pi^2/32 \approx 0.308425$ from below, consistent with the $O(K^{-2})$ correction term found in the proof.

Table 92.1: Toy trade-off between distortion cost $D(K)$ and efficiency cost $E(K) = K$ (Definition 92.1), verifying the Pareto relation Eq. (92.1). Source: `code/tradeoff_toy_model.py`, `data/tradeoff_toy_model_results.json`.

| $K$ | $D(K)$ | $E(K)$ | $D(K)\cdot E(K)^2$ |
|---|---|---|---|
| 4 | $1.9215\times10^{-2}$ | 4 | 0.30744 |
| 8 | $4.8153\times10^{-3}$ | 8 | 0.30818 |
| 16 | $1.2045\times10^{-3}$ | 16 | 0.30836 |
| 32 | $3.0118\times10^{-4}$ | 32 | 0.30841 |
| 64 | $7.5298\times10^{-5}$ | 64 | 0.30842 |
| 128 | $1.8825\times10^{-5}$ | 128 | 0.30842 |
| 256 | $4.7062\times10^{-6}$ | 256 | 0.30842 |

**Key Result**

Within this toy model, distortion cost and efficiency cost cannot both be driven to zero: their product $D(K)\cdot E(K)^2$ tends to a strictly positive constant, $\pi^2/32$, rather than to zero. This is precisely the *qualitative* shape of tension Wecker et al. (2024) describe in words — a real trade-off, not merely two costs that both happen to be nonzero at any finite $K$. The specific exponent (quadratic in $E(K)$) and the specific constant ($\pi^2/32$) are artefacts of this toy model's particular choice of $D$ and $E$, not a claim about any real DGGS implementation.

## 92.3 Scope and what a real trade-off analysis would need

**To Complete / Verify**

A non-toy version of this chapter would need: (i) a distortion cost measured on an actual DGGS cell geometry (H3, S2, rHEALPix, DGGRID, OpenEAGGR, or Wecker et al.'s MBD itself), reusing Kmoch et al. (2022)'s normalized-area and compactness metrics once Section 1.3's reproduction task is complete; and (ii) an efficiency cost measured in an operational unit relevant to real indexing and query workloads (e.g. cell-lookup time, neighbour-query complexity, or storage overhead), rather than the stylized ring count $E(K) = K$ used here. Both are left for future work; this chapter establishes only that the qualitative trade-off Wecker et al. (2024) describe has a clean, exactly provable analogue once a distortion and an efficiency cost are both defined precisely on the *same* tessellation family, which is itself a nontrivial methodological point worth keeping when the real (non-toy) comparison is eventually attempted.

*Update:* Chapter 93 carries out exactly this comparison for H3, reusing Chapter 12's independently reproduced cell geometry for the distortion side and a directly measured index-storage cost for the efficiency side.

Chapter 93

# A Real (Non-Toy) Trade-off: Distortion Against Operational Cost on H3

## 93.1 Scope: items (i) and (ii) of the toy chapter's todobox

Section 92.3 left two items open: (i) a distortion cost measured on an actual DGGS cell geometry rather than the toy lat/lon graticule, and (ii) an efficiency cost measured in an operational unit relevant to real indexing and query workloads, rather than the stylized ring count $E(K) = K$. This chapter attempts both, on H3 specifically, reusing *exactly* the method and code already verified in Chapter 12 (`code/independent_h3_reproduction.py`) rather than inventing a new distortion metric, and measuring the efficiency side directly from `h3-py`'s own index representation rather than assuming a figure for it.

**Methodological Note**

**What was done, precisely.** H3 resolutions 0–5 were regenerated in full (globally, all base cells and their children) using `h3-py` v4, with cell area from `h3-py`'s own `cell_area()` and perimeter from the same hand-rolled great-circle routine as Chapter 12 — identical methodology, extended from that chapter's two resolutions {2,3} to six, {0,...,5}. Resolution 5 (2,016,842 cells) was included because it remained tractable to generate in full (28 seconds of wall-clock time); resolution 6 (over 14 million cells) was not attempted. For the efficiency side, each resolution's H3 index was inspected directly: `h3.latlng_to_cell(lat, lon, res)` returns a fixed-width hexadecimal string whose integer value has bit-length 60 at *every* resolution tested, 0 through 5 — confirming empirically, rather than merely citing H3's documentation, that an H3 index occupies a fixed 64-bit word regardless of resolution. The operational efficiency cost used below is therefore $E(\text{res}) = \texttt{num_cells}(\text{res}) \times 64$ bits: the storage required to hold one index per cell for a full global tessellation at that resolution — a real, load-bearing quantity for any application that must enumerate, index, or persist a global DGGS layer, not a stylized proxy. Full code: `code/real_tradeoff_analysis.py`; full output: `data/real_tradeoff_analysis_results.json`.

## 93.2 The extended table

**Key Result**

The two costs move on entirely different clocks. Efficiency cost $E(\text{res})$ grows geometrically, by very close to the expected factor of 7 per level (H3's aperture-7 hierarchy), across the whole range: $122 \to 842 \to 5{,}882 \to 41{,}162 \to 288{,}122 \to 2{,}016{,}842$ cells, a $16{,}531$-fold increase in storage from resolution 0 to resolution 5. Distortion cost $D(\text{res}) = \sigma(\text{norm. area})$ does the opposite: it drops sharply once, from $0.1478$ at resolution 0 to $0.1356$ at resolution

Table 93.1: H3, resolutions 0–5: distortion cost $\sigma$(norm. area) (independently computed, Chapter 12's method) against operational efficiency cost, measured as cell count and as total index storage at a fixed, empirically confirmed 64-bit index width. Source: `code/real_tradeoff_analysis.py`.

| res | $\sigma$(norm. area) | #cells | storage (bits) | storage (MiB) |
|---|---|---|---|---|
| 0 | 0.14782 | 122 | 7,808 | 0.00093 |
| 1 | 0.13561 | 842 | 53,888 | 0.00642 |
| 2 | 0.13373 | 5,882 | 376,448 | 0.04486 |
| 3 | 0.13346 | 41,162 | 2,634,368 | 0.31394 |
| 4 | 0.13342 | 288,122 | 18,439,808 | 2.19782 |
| 5 | 0.13342 | 2,016,842 | 129,077,888 | 15.38475 |

1 (an 8.3% relative reduction for a 7× storage increase), drops much less from resolution 1 to 2 ($0.1356 \to 0.1337$, a 1.4% reduction for another 7×), and from resolution 3 onward is flat to four significant figures $(0.13346, 0.13342, 0.13342)$: a further 16-fold increase in storage from resolution 3 to 5 buys a reduction in $\sigma$ of 0.00004, i.e. three-hundredths of one percent.

*Remark* 93.1 (This independently confirms, and sharpens, Chapter 13). Chapter 13 already found, from Kmoch & Vasilyev (2022)'s *transcribed* statistics at resolutions 2–4, that H3's $\sigma$ was flat to under 4% drift. The present chapter's independently computed figures (a different code path entirely: own tessellation generation, own perimeter routine, own resolution range) agree with that finding and extend it in the direction that matters most for a trade-off argument: resolutions 0 and 1, which Chapter 13 did not have data for, are exactly where almost all of the *genuine* distortion-reduction benefit of refining the DGGS turns out to live. From resolution 2 onward, refining H3 further is, on this measure, close to a pure efficiency cost with no distortion dividend.

## 93.3 What this does, and does not, resolve about the toy trade-off

Chapter 92's toy model built $D(K)$ and $E(K)$ so that increasing $K$ always buys a strictly positive, if diminishing, reduction in $D$ — reproducing, in stylized form, the qualitative shape Wecker et al. (2024) describe (equal-area constructions cost more but distort less, and the two costs trade off smoothly). The real H3 data above show that this qualitative picture, while faithful to the tension Wecker et al. (2024) state *in general terms* for DGGS design as a category, does *not* describe how H3's own resolution parameter behaves once measured directly: for H3 specifically, distortion cost is not a smoothly decreasing function of efficiency cost across the whole range, but a function that decreases sharply over two levels and then saturates, while efficiency cost keeps growing geometrically without bound. This is not a criticism of Wecker et al. (2024), whose proposed mesh-based DGGS is a different construction addressing a different resolution mechanism (subdivision of the base polyhedron, not H3's aperture-7 hierarchical rings); it is a concrete illustration of exactly the gap the toy chapter's todobox flagged, between a stylized qualitative trade-off and what a specific, real DGGS implementation actually does when the two costs are defined precisely and measured on the same tessellation family.

**To Complete / Verify**

This closes items (i) and (ii) of Section 92.3 for H3 only. Three things remain open. First, whether the same sharp-drop-then-saturate pattern holds for S2, rHEALPix, DGGRID, or OpenEAGGR — Chapter 13 found a qualitatively different pattern for S2 (a slower, later saturation near $\sigma \approx 0.152$, not H3's near-immediate one near $\sigma \approx 0.133$), which is suggestive but was established there only on Kmoch & Vasilyev (2022)'s transcribed statistics, not on an independently reproduced S2 tessellation. Second, storage in bits is only one operational cost among several named in Section 92.3 (cell-lookup time and neighbour-query complexity were not attempted here, and would require timing actual `h3-py` operations rather than counting cells).

*Update:* Chapter 94 times exactly these two operations directly. Third, no attempt is made here to fit $D(\text{res})$ and $E(\text{res})$ into a single constrained-optimization statement comparable in form to Chapter 92's toy Pareto proposition (Proposition 92.2); the data above support a qualitative diagnosis (saturation) but not yet a closed-form successor to that proposition.

Chapter 94

# Beyond Storage: Timing the Operations Themselves

## 94.1 Scope: the second open item of the real-tradeoff chapter

Chapter 93's todobox named storage as only one operational cost among several, and left cell-lookup time and neighbour-query complexity unmeasured. This chapter measures them directly: not as a theoretical complexity claim transcribed from H3's documentation, but as timed wall-clock measurements of actual `h3-py` calls, executed directly on this machine.

**Methodological Note**

**What was done, precisely, and its limits.** Three operations were timed at each resolution 0–5: `latlng_to_cell` (coordinate → index encoding), `cell_to_parent` (one step up the hierarchy; undefined at resolution 0), and `grid_disk(…, 1)` (the 1-ring of immediate neighbours, the basic primitive of any neighbour-query workload). For each operation and resolution: a discarded warm-up of 2,000 calls, then 21 timed batches of 20,000 calls each, with the reported figure the *median* of the 21 per-call batch times (median rather than mean, to resist single scheduling-jitter outliers — standard microbenchmark practice). This was run twice, independently; the two runs agree qualitatively (same ordering, same flat-versus-growing pattern across resolutions) but not to more than roughly two significant figures in absolute microseconds, which is expected: this is a single-core container (`nproc`=1, Intel Xeon @ 2.80GHz, Python 3.12.3, `h3-py` 4.5.0), not a controlled benchmarking rig, and absolute timings on such a machine are not portable to any other hardware, JIT state, or system load. What *is* reported below as a finding is only the pattern that survived both runs, not any single absolute number. Full code: `code/h3_operational_timing.py`; full output (second run, the one tabulated below): `data/h3_operational_timing_results.json`.

## 94.2 The measured table

**Key Result**

The three operations show three different relationships to resolution, none of them resembling Chapter 93's storage cost (a 16,531-fold increase from resolution 0 to 5). `cell_to_parent` is flat within measurement noise (0.63–0.67 $\mu$s, no monotonic trend across six resolutions): a single hierarchy step costs the same regardless of how deep the hierarchy is, consistent with H3's design as a fixed-digit positional index rather than a pointer chain that would need to be walked. `grid_disk(k=1)` is likewise flat (5.9–6.4 $\mu$s, no trend): finding a cell's six or seven immediate neighbours costs the same at resolution 5 as at resolution 0. `latlng_to_cell` is the one operation with a visible, monotonic trend:

Table 94.1: Median per-call time (microseconds) for three H3 operations across resolutions 0–5, second of two independent timing runs. Source: `code/h3_operational_timing.py`, `data/h3_operational_timing_results.json`.

| res | `latlng_to_cell` | `cell_to_parent` | `grid_disk(k=1)` |
|---|---|---|---|
| 0 | 0.628 | — | 6.293 |
| 1 | 0.655 | 0.669 | 6.215 |
| 2 | 0.662 | 0.654 | 5.892 |
| 3 | 0.676 | 0.627 | 5.932 |
| 4 | 0.685 | 0.664 | 6.032 |
| 5 | 0.691 | 0.647 | 6.377 |

$0.628 \rightarrow 0.691\,\mu$s from resolution 0 to 5, an increase of about 10% over the whole range, in both independent runs. This is the expected shape for an operation that must encode one base-7 digit per resolution level into the index (six extra digits at resolution 5 versus resolution 0): weakly growing in resolution, but by nowhere near the $16{,}531\times$ growth of the storage cost measured in Chapter 93.

*Remark* 94.1 (What this adds to the real trade-off). Chapter 93 showed that H3's distortion cost saturates while its storage cost grows geometrically. This chapter shows that a third quantity — per-operation query cost, for the two operations that matter most for actually *using* a DGGS layer once built — does not track storage cost at all: it is flat, or nearly flat, across the same resolution range. The practical implication, stated carefully: for an application whose bottleneck is query latency rather than total index storage, refining H3 resolution past the point where distortion saturates (resolution 2–3, per Chapter 93) is not obviously penalized on the query-cost axis the way it clearly is on the storage axis. Whether it is penalized on *other* axes not measured here (index-build time for a full global layer, memory locality effects on real disk-backed storage rather than a Python dictionary, or serialization cost) is not established by this chapter and should not be inferred from it.

## 94.3 What remains open

**To Complete / Verify**

This measures three operations, on one library, on one single-core machine, in one Python process — it is a first empirical pass, not a benchmark suite. Chapter 96 has since repeated this for S2: two of its three operations reproduce the flat pattern found here (at different absolute constants), the third does not. Left open: whether the same holds for rHEALPix, DGGRID, or OpenEAGGR (each has a different hierarchy encoding and could plausibly show a different pattern); whether it holds for larger neighbourhoods ($k > 1$ in `grid_disk`); whether it holds under concurrent/multi-core load rather than a single sequential process; and whether a compiled-language (non-Python) binding to the same underlying H3 C library would show the same relative pattern with different absolute constants, or a different pattern altogether. None of these was attempted here.

Chapter 95

# The Same Real Trade-off, on S2

## 95.1 Scope: the first item of Chapter 93's closing todobox

Chapter 93 closed by naming three open items. The first was left explicitly unresolved: whether the sharp-drop-then-saturate pattern found there for H3's distortion cost holds for S2 as well, or whether Chapter 13's finding of a qualitatively different pattern in Kmoch & Vasilyev (2022)'s *transcribed* statistics — a slower, later saturation near $\sigma \approx 0.152$ rather than H3's near-immediate one near $\sigma \approx 0.133$ — survives an independent, from-scratch reproduction. This chapter attempts exactly that, reusing the method already verified for S2 in Chapter 14 (native `s2sphere` enumeration, full global sweep, no sampling) rather than any part of Kmoch & Vasilyev (2022)'s own pipeline.

**Methodological Note**

**What was done, precisely.** For each S2 level $L = 0, \ldots, 8$, every cell at that level was enumerated natively via `s2sphere` v0.2.5 (`CellId.begin(L)` stepped with `.next()` to `CellId.end(L)`), giving the full, unfiltered global count $6 \cdot 4^L$ — the same enumeration already used in Chapter 14, extended downward to levels 0–1, which that chapter did not cover, specifically to check whether an H3-like early sharp drop is visible at the coarsest S2 levels. Area came from `s2sphere`'s own `Cell.exact_area()` — the library's own spherical-excess engine, exactly analogous to how Chapters 12 and 93 use `h3-py`'s own `cell_area()` rather than a hand-rolled formula. Perimeter is the sum of the four edge angles between consecutive cell vertices, computed via `s2sphere`'s own `Point.angle()` on the unit-sphere vertex vectors — the S2 analogue of the H3 scripts' hand-rolled haversine perimeter, using the library's native vector geometry instead of a re-derivation, since S2 vertices are already unit vectors rather than lat/lon pairs. $\sigma$(norm. area) is computed identically to Chapter 12: each cell's area divided by the mean area at that level, then the population standard deviation of that ratio. Level 8 (393,216 cells) was the practical ceiling for generating the full level in reasonable wall-clock time (10.2 seconds); level 9 (1,572,864 cells) was not attempted. For the efficiency side, each level's S2 `CellId.id()` was inspected directly, exactly as Chapter 93 did for H3's index: the measured integer bit-length is 63 at every level tested, 0 through 8, confirming empirically that an S2 cell ID also occupies (at most) a fixed 64-bit word, like H3's. Full code: `code/s2_real_tradeoff_analysis.py`; full output: `data/s2_real_tradeoff_analysis_results.json`.

Table 95.1: S2, levels 0–8: distortion cost $\sigma$(norm. area) (independently computed, native full enumeration, no data-line filtering) against operational efficiency cost, measured as cell count and as total index storage at a fixed, empirically confirmed 64-bit index width. Source: `code/s2_real_tradeoff_analysis.py`.

| level | $\sigma$(norm. area) | #cells | storage (bits) | storage (MiB) |
|---|---|---|---|---|
| 0 | 0.00000 | 6 | 384 | 0.0000458 |
| 1 | 0.00000 | 24 | 1,536 | 0.000183 |
| 2 | 0.07956 | 96 | 6,144 | 0.000732 |
| 3 | 0.13827 | 384 | 24,576 | 0.00293 |
| 4 | 0.15039 | 1,536 | 98,304 | 0.01172 |
| 5 | 0.15335 | 6,144 | 393,216 | 0.04688 |
| 6 | 0.15408 | 24,576 | 1,572,864 | 0.1875 |
| 7 | 0.15426 | 98,304 | 6,291,456 | 0.75 |
| 8 | 0.15431 | 393,216 | 25,165,824 | 3.0 |

## 95.2 The table

**Key Result**

S2's distortion cost does not drop with refinement, as H3's does; it *rises* and then saturates. At levels 0 and 1, every cell is one of the 6 (or 24) top-level face subdivisions and is, to double-precision floating point, exactly area-uniform: $\sigma = 0.0$ identically (the only nonzero digits at level 1, $2.48 \times 10^{-16}$ in the raw output, are floating-point noise, not signal). $\sigma$ then rises sharply and monotonically from level 2 onward — $0.07956 \to 0.13827 \to 0.15039 \to 0.15335$ across levels 2–5 — before flattening: levels 6, 7, and 8 give $0.15408$, $0.15426$, $0.15431$, moving by only $0.00023$ and then $0.00005$, against storage that keeps growing by exactly $4\times$ per level throughout the whole range (a $65{,}536$-fold increase, $6 \to 393{,}216$ cells, from level 0 to level 8, exactly $4^8$). This is the opposite shape from H3's: H3 starts near its most distorted and improves fastest at the coarsest levels tested; S2 starts perfectly uniform at its coarsest levels and *becomes* distorted as it refines, before the two curves converge on a similarly-shaped saturation plateau.

*Remark* 95.1 (Independent cross-check against Chapter 13's transcribed figures). Chapter 13 reported, from Kmoch & Vasilyev (2022)'s transcribed summary statistics, $\sigma = 0.15083$, $0.15148$, $0.15167$, $0.15172$ at S2 resolutions 5, 6, 7, 8 (using Kmoch & Vasilyev's own date-line-adjusted cell counts: 6,017, 24,201, 97,065, 388,775, each a little below this chapter's naive $6 \cdot 4^L$ counts, per Chapter 14's deficit table). The present, independently computed figures – $0.15335$, $0.15408$, $0.15426$, $0.15431$ at the same four levels – agree on every qualitative point: rising then saturating, converging toward a value in the same narrow neighbourhood just above $0.15$. They do not agree to the precision Remark 93.1 found for H3: each of the four levels here reads $0.0018$–$0.0026$ higher than Kmoch & Vasilyev's corresponding figure, a small but consistent offset in the same direction at every level. Two candidate explanations are visible from within this volume's own data and neither is confirmed here: this chapter's counts are the *naive* $6 \cdot 4^L$ enumeration, including the date-line-crossing cells that Chapter 14 found Kmoch & Vasilyev's own pipeline excludes (a few hundred cells out of tens or hundreds of thousands at these levels); and this chapter's perimeter uses `s2sphere`'s vertex-angle geometry rather than whatever perimeter routine underlies Kmoch & Vasilyev's $\sigma$, which Chapter 14 never needed to reproduce. Distinguishing between these two candidate sources of the offset — or finding a third – is not attempted here and is left as an open item below, in the same spirit as Chapter 14's own two-mechanism, not-fully-resolved diagnosis of the resolution-to-level deficit.

## 95.3 Diagnosing the offset: ruling out date-line-crossing inclusion

Remark 95.1 named two candidate mechanisms for the offset without testing either. This section tests the first: that this chapter's naive $6 \cdot 4^L$ enumeration includes date-line-crossing cells that Kmoch & Vasilyev's own pipeline excludes (Chapter 14).

**Methodological Note**

**What was done, precisely.** Section 95.2's own computation was re-run at levels 5–8, unchanged in every respect (same cells, same `exact_area()` values, same vertex-angle perimeters), except that cells flagged by Chapter 14's own `crosses_dateline()` test (a jump of more than 180° in longitude between consecutive boundary vertices, applied to the same four vertices already needed for the perimeter) were excluded before recomputing $\sigma$(norm. area). This is the same exclusion rule already verified in `code/s2_resolution_ambiguity.py`, applied here for the first time to the distortion statistic rather than only to the cell count. Full code: `code/s2_real_tradeoff_dateline_filtered.py`; full output: `data/s2_real_tradeoff_dateline_filtered_results.json`.

Table 95.2: Levels 5–8: naive $\sigma$, $\sigma$ after excluding date-line-crossing cells, Kmoch & Vasilyev (2022)'s transcribed $\sigma$, and the offset against it before and after filtering. Source: `data/s2_real_tradeoff_dateline_filtered_results.json`.

| level | $\sigma$ naive | $\sigma$ filtered | Kmoch & Vasilyev | offset (naive) | offset (filtered) |
|---|---|---|---|---|---|
| 5 | 0.15335 | 0.15332 | 0.15083 | 0.00252 | 0.00249 |
| 6 | 0.15408 | 0.15403 | 0.15148 | 0.00260 | 0.00255 |
| 7 | 0.15426 | 0.15423 | 0.15167 | 0.00259 | 0.00256 |
| 8 | 0.15431 | 0.15429 | 0.15172 | 0.00259 | 0.00257 |

**Key Result**

Date-line-crossing inclusion is *not* the explanation. Excluding those cells — only 65 to 513 out of tens or hundreds of thousands, 1.1% of the total at level 5 falling to 0.13% at level 8 — moves $\sigma$ by at most 0.00005 at any of the four levels tested, shrinking the offset against Kmoch & Vasilyev's figures by less than 0.0001 throughout, leaving 99 % or more of the original 0.0025–0.0026 gap unexplained at every level. The first candidate mechanism named in Remark 95.1 is therefore ruled out, cleanly and by a wide margin, rather than confirmed. This is reported as a negative result with the same visibility as this volume's positive ones, following Chapter 114's stated discipline.

By elimination, this sharpens — but does not yet confirm — the second candidate: a perimeter-routine difference between this chapter's vertex-angle method and whatever underlies Kmoch & Vasilyev's own $\sigma$. Since $\sigma$(norm. area) depends only on *area*, not perimeter, a perimeter difference cannot be the mechanism either; Kmoch & Vasilyev's $\sigma$ must therefore differ from this chapter's on the *area* side, not the perimeter side, which Remark 95.1 did not distinguish. Two remaining candidates, neither tested here: a different area engine (Kmoch & Vasilyev's S2 binding may not use `exact_area()`'s spherical-excess computation), or a difference in which cells are included beyond date-line-crossing ones (Chapter 14's own polar-cap mechanism, reverse-fit rather than derived, was never confirmed as the true remaining cause of the *count* deficit either, and could equally affect $\sigma$ if a systematically atypical set of cells near the poles is excluded).

## 95.4 What this resolves, and what it does not, about Chapter 93's open item

The qualitative claim Chapter 13 could only make from transcribed, secondary-source figures — that S2 saturates more slowly and at a higher plateau than H3, rather than sharing H3's near-immediate saturation — now stands on an independent, from-scratch computation using this volume's own code path, and is confirmed: S2's $\sigma$ is still changing at the fourth significant figure between levels 7 and 8 ($0.15426 \rightarrow 0.15431$), a full two levels past where H3's $\sigma$ had already stabilized to four significant figures (Table 93.1, resolutions 3–5). What this chapter adds, beyond confirmation, is the shape of S2's curve at the levels Kmoch & Vasilyev's dataset did not cover: not a smoothly decreasing distortion cost extending H3's pattern downward to coarser levels, but a curve that starts at exactly zero at levels 0–1 and *increases* through level 5 before flattening — a qualitatively different low-end behaviour from H3's, not merely a slower version of the same one. Storage cost, in contrast, behaves identically in kind to H3's: geometric growth with no saturation, at a smaller per-level factor ($4\times$ against H3's $7\times$, reflecting the two DGGS's different aperture) but the same unbounded shape.

**To Complete / Verify**

This closes the first item of Chapter 93's closing todobox for S2. Two things remain open. First, the small but systematic $0.0018$–$0.0026$ offset against Kmoch & Vasilyev's transcribed figures (Remark 95.1) is now narrowed rather than fully diagnosed: Section 95.3 rules out date-line-crossing inclusion (it accounts for under 2% of the gap) and argues, from $\sigma$(norm. area)'s definition, that the remaining explanation must lie on the area-computation side rather than the perimeter side — but does not identify which area engine or cell-selection difference is responsible, and does not rule out Chapter 14's own unconfirmed polar-cap mechanism as a contributor; this narrower question is the sole item still open from this chapter.
**Update:** The second item is closed. Chapter 97 subsequently reproduced rHEALPix independently on both distortion and storage; Chapter 105 and Chapter 108 did the same for DGGRID and OpenEAGGR respectively, so all five DGGS this volume treats are now measured by its own code on both axes. Chapter 96 likewise repeated Chapter 94's query-cost timing (neighbour lookup, hierarchy traversal) for S2 directly. Only the area-computation offset above remains unresolved.

Chapter 96

# Timing S2's Operations, the Same Way

## 96.1 Scope: the second open item of the S2 real-tradeoff chapter

Chapter 95's closing todobox left a second item open: Chapter 94's query-cost timing, done for H3, had not been repeated for S2. This chapter does that, timing the direct S2 analogues of the same three H3 operations, with the same method and the same disclosed limits.

**Methodological Note**

**What was done, precisely, and its limits.** Three operations were timed at each S2 level 0–8: `CellId.from_lat_lng(…).parent(level)` (coordinate → index encoding at the level of interest, S2's own idiom since `from_lat_lng` alone returns a leaf-level id), `cell.parent(level - 1)` (one step up the hierarchy; undefined at level 0), and `list(cell.get_all_neighbors(level))` (the same-level neighbour set — 8 cells for S2's quadtree adjacency, not H3's 6–7-cell hex ring; the two counts are not commensurable and no claim is made that they should be). Timing method identical to Chapter 94: a discarded warm-up of 2,000 calls, then 21 timed batches of 20,000 calls each, median per-call time reported. Run twice, independently, on the same machine as Chapter 94 (single-core container, `nproc`=1, Intel Xeon @ 2.80GHz, Python 3.12.3, `s2sphere` 0.2.5); absolute timings are not portable beyond this environment, and only the pattern that survived both runs is reported as a finding. Full code: `code/s2_operational_timing.py`; full output (second run, the one tabulated below): `data/s2_operational_timing_results_run2.json` (first run: `data/s2_operational_timing_results_run1.json`).

## 96.2 The measured table

Table 96.1: Median per-call time (microseconds) for three S2 operations across levels 0–8, second of two independent timing runs. Source: `code/s2_operational_timing.py`, `data/s2_operational_timing_results_run2.json`.

| level | `from_lat_lng.parent` | `parent` | `get_all_neighbors` |
|---|---|---|---|
| 0 | 7.382 | — | 63.783 |
| 1 | 7.139 | 1.138 | 55.669 |
| 2 | 7.324 | 1.225 | 50.122 |
| 3 | 7.177 | 1.152 | 49.802 |
| 4 | 7.236 | 1.172 | 46.950 |
| 5 | 7.356 | 1.162 | 47.069 |
| 6 | 7.411 | 1.209 | 49.415 |
| 7 | 7.574 | 1.229 | 48.073 |
| 8 | 7.271 | 1.171 | 40.030 |

**Key Result**

Two of the three operations reproduce Chapter 94's flat pattern for H3, at different absolute constants. `parent` is flat within measurement noise across both runs (1.14–1.23 $\mu$s, no monotonic trend), like H3's `cell_to_parent` (0.63–0.67 $\mu$s): a single hierarchy step costs the same regardless of depth, for both DGGS's positional-index designs. `from_lat_lng.parent` (the encoding operation) is also essentially flat (7.14–7.57 $\mu$s across both runs, no monotonic trend), unlike H3's own encoding operation, which showed a small but real 10% monotonic increase over the same kind of range (Table 94.1): S2's 64-bit Hilbert-curve-style id is written by a single fixed-cost bit-interleaving step regardless of level, whereas H3's per-digit base-7 encoding must write one more digit per additional resolution level, a structural difference between the two indexing schemes that this flat-versus-rising contrast makes newly visible. The third operation, `get_all_neighbors`, does *not* reproduce H3's flat pattern: it falls from 63.8 $\mu$s at level 0 to a plateau around 47–50 $\mu$s by level 4, holds there through level 7, then drops again to 40.0 $\mu$s at level 8 — a shape confirmed in both independent runs (run 1: $65.5 \to 55.5 \to 48.9 \to \cdots \to 40.4$ $\mu$s; run 2 essentially the same numbers to within a few percent at every level), not a single-run artifact.

*Remark* 96.1 (A new, unexplained pattern, disclosed rather than smoothed over). The falling-then-plateauing shape of `get_all_neighbors` has no counterpart in this volume's H3 timings and is not explained by anything measured here. Chapter 14's own investigation found S2's face structure and polar-region cell geometry to behave differently near the poles at low levels than at high ones, which is at least consistent with a level-dependent cost for a neighbour-finding routine that must handle face-boundary and polar-region special cases — but this is a plausible association, not a tested mechanism, and no attempt is made here to isolate face-boundary or polar cells' timing from the rest. The one thing established with confidence is that the shape is real (both runs agree) and is specific to `get_all_neighbors`: the other two operations, which do not need to reason about cell adjacency, are flat.

## 96.3 What this adds to the S2 real trade-off

Chapter 95 showed that S2's distortion cost rises then saturates while its storage cost grows by exactly $4\times$ per level, without bound. This chapter shows that, as with H3 (Remark 94.1), per-operation query cost does not track storage cost: two of three operations are flat across the full level range, and the third falls or holds rather than growing. For an application bottlenecked by query latency rather than storage, refining S2 past its distortion-saturation point (around level 5, Chapter 95) carries, on this evidence, no obvious query-cost penalty — if anything, `get_all_neighbors` becomes marginally *cheaper* at the higher levels tested. This should be read with the same caution Remark 94.1 attached to the H3 finding: index-build time, disk-backed memory locality, and serialization cost are not measured here and should not be inferred from query latency alone.

**To Complete / Verify**

This closes the second item of Chapter 95's closing todobox. What remains open: the mechanism behind `get_all_neighbors`'s falling-then-plateauing shape (Remark 96.1) is disclosed but not identified; whether the same shape appears for larger neighbourhoods (S2 has no single direct analogue of H3's `grid_disk(k>1)` tested here); whether rHEALPix, DGGRID, or OpenEAGGR show flat, rising, or falling query-cost patterns remains entirely untested by this volume; and, as for H3, whether any of these patterns survive under con-

current load or on a compiled-language (non-Python) binding to the same underlying S2 library is not attempted here.

Chapter 97

# A Third DGGS, and a Structurally Different Trade-off: rHEALPix

## 97.1 Scope: closing part of Chapter 11's oldest open item

Section 11.5's item (i), left open since Chapter 11 and repeated in every closing todobox since, named a fully independent reproduction — regenerating cell geometry from a software package independent of Kmoch & Vasilyev (2022)'s own files — as attempted for H3 only (Chapter 12), with S2, rHEALPix, DGGRID, and OpenEAGGR explicitly listed as not attempted. Chapter 95 closed the S2 half of that gap. This chapter closes the rHEALPix half, using the `rhealpixdggs` 0.6.0 Python package, a codebase independent of both Kmoch & Vasilyev (2022)'s own pipeline and of `h3-py`/`s2sphere`.

**Methodological Note**

**What was done, precisely.** Every cell at resolutions 0 through 5 was enumerated natively via `RHEALPixDGGS.grid(r)` (N_side= 3, the package default), giving the full global count $6 \cdot 9^r$ $(6, 54, 486, 4{,}374, 39{,}366, 354{,}294)$. For each cell: area via `Cell.area(plane=False)` — the ellipsoid/sphere area, *not* the library's default `plane=True` planar (projected) area, which would be the wrong quantity for a sphere/ellipsoid distortion comparison and was corrected for here; perimeter as the geodesic quadrilateral connecting the four reported corner vertices (`Cell.vertices(plane=False)`) via great-circle (haversine) arcs, the same corner-to-corner approximating convention already used for H3 and S2 in this volume, disclosed here too as an approximation to rHEALPix's true (non-geodesic, authalic-projection) cell boundary rather than its exact length. Level 5 (354,294 cells) took 84.7 seconds, the practical ceiling for wall-clock generation time; level 6 (3,188,646 cells) was not attempted. Full code: `code/rhealpix_real_tradeoff_analysis.py`; full output: `data/rhealpix_real_tradeoff_results.json`.

## 97.2 Area distortion: testing the "equal-area" name directly

**Key Result**

rHEALPix cells are exactly, not approximately, equal in area, at every resolution tested. The number of *distinct* area values among all cells at a resolution — 6 cells at resolution 0, up to 354,294 at resolution 5 — was counted directly rather than a standard deviation estimated: it is exactly 1 at every resolution, 0 through 5. $\sigma$(norm. area) is therefore not merely small, as Table 11.1's transcribed 0.00193–0.00201 figures for rHEALPix already showed relative to H3 and S2; on this volume's own from-scratch computation it is exactly zero, to full floating-point precision, at both resolutions that table covers and at three further ones besides. This is a mathematical property of the HEALPix construction (equal-area

*by* construction, not an empirical regularity that happens to hold), confirmed here rather than assumed from the name.

*Remark* 97.1 (The transcribed 0.002 figure does not survive an ellipsoid check either). Table 11.1 reported, from Kmoch & Vasilyev (2022)'s transcribed statistics, $\sigma = 0.00193$ (resolution 2) and 0.00201 (resolution 4) for rHEALPix — small, but not zero. One candidate explanation was tested directly: that the transcribed figures used a real ellipsoid (WGS84) rather than a perfect sphere, and that rHEALPix's equal-area guarantee is only exact on the authalic sphere, degrading slightly once mapped onto an actual ellipsoid. This was tested by re-running Section 97.2's own computation at resolutions 2 and 4 with `rhealpixdggs`'s `WGS84_ELLIPSOID` in place of the unit sphere: the result is unchanged, exactly 1 distinct area value at both resolutions, $\sigma = 0$ again. The ellipsoid is not the explanation, at least not via this package's own area computation. What remains unexplained, and is not resolved here: the transcribed cell counts themselves do not match this chapter's either (453 against this chapter's 486 at "resolution 2", 38,731 against 39,366 at "resolution 4" — deficits of 33 and 635 cells respectively) which raises the same possibility already documented for S2 in Chapter 14: that Kmoch & Vasilyev's "rHEALPix resolution" label does not correspond one-to-one with `rhealpixdggs`'s own resolution parameter, whether from a different `N_side`, a different face-layout convention, or a filtering step this chapter's naive enumeration does not apply. If the underlying cell populations differ by three or six hundred cells, their area statistics computed by two different pipelines need not agree exactly even where both correctly implement an equal-area scheme on their own respective cell sets. This is named as an open rHEALPix resolution-correspondence question, structurally analogous to Chapter 14's S2 one, and is not investigated further here.

## 97.3 Shape distortion: the isoperimetric quotient

Because area distortion is structurally absent, this chapter measures the distortion cost that *is* present: cell shape, via the isoperimetric quotient already used as a secondary quantity in Chapters 93 and 95 (ipq $= 1$ for a circle, $\pi/4 \approx 0.7854$ for a square).

Table 97.1: rHEALPix, resolutions 0–5: cell count, confirmed distinct area values (equal-area check), and isoperimetric quotient statistics. Source: `code/rhealpix_real_tradeoff_analysis.py`.

| res | #cells | distinct areas | IPQ mean | IPQ std | IPQ min | IPQ max |
|---|---|---|---|---|---|---|
| 0 | 6 | 1 | 1.10911 | 0.15936 | 0.99642 | 1.33448 |
| 1 | 54 | 1 | 0.79437 | 0.09389 | 0.72446 | 1.24163 |
| 2 | 486 | 1 | 0.75807 | 0.05352 | 0.63598 | 1.23455 |
| 3 | 4,374 | 1 | 0.75388 | 0.04899 | 0.60852 | 1.23379 |
| 4 | 39,366 | 1 | 0.75340 | 0.04884 | 0.59532 | 1.23371 |
| 5 | 354,294 | 1 | 0.75335 | 0.04887 | 0.58900 | 1.23370 |

**Key Result**

IPQ follows an H3-like early-saturation curve, not an S2-like rising one: the mean falls sharply from resolution 0 to resolution 3 ($1.10911 \to 0.79437 \to 0.75807 \to 0.75388$) and is then essentially flat through resolution 5 ($0.75340 \to 0.75335$, a change of 0.00005 across two full resolution levels and a $9\times$ increase in cell count). The spread (IPQ std) shows the same shape: $0.15936 \to 0.09389 \to 0.05352 \to 0.04899$, saturating near 0.0488–0.0489 by resolution 3. The resolution-0 mean above 1 (the theoretical ceiling for a true IPQ)

is a visible artifact of the corner-to-corner geodesic approximation at the coarsest resolution, where each of the 6 cells covers a quarter of a hemisphere and the true (curved, non-geodesic) cell boundary is poorly approximated by four great-circle chords; it is not evidence of a genuinely super-circular shape and is reported as an approximation artifact, not a finding about rHEALPix geometry.

*Remark* 97.2 (A closer cross-check than the area one). Table 11.1's companion IPQ table (not reproduced in full here) gave Kmoch & Vasilyev (2022)'s transcribed rHEALPix figures as mean 0.75693 (std 0.04257) at resolution 2 and mean 0.75394 (std 0.04798) at resolution 4. Against this chapter's own 0.75807 (std 0.05352) at resolution 2 and 0.75340 (std 0.04884) at resolution 4: the resolution-4 means agree to within 0.0005 and the stds to within 0.0009 — still closer than the area cross-check (Remark 97.1) or the S2 cross-check (Remark 95.1) managed for their respective quantities, despite *this* chapter's admittedly cruder straight-chord perimeter approximation. The resolution-2 comparison agrees less closely (means within 0.0011, stds within 0.011), consistent with the cell-count mismatch (Remark 97.1) being larger, proportionally, at resolution 2 ($33/486 \approx 6.8\%$) than at resolution 4 ($635/39{,}366 \approx 1.6\%$): a shape statistic computed over a population differing by a larger fraction of cells would be expected to agree less well, for reasons having nothing to do with either pipeline's geometric correctness.

## 97.4 Storage cost, measured without an assumed fixed-width index

Unlike H3 and S2, `rhealpixdggs` exposes no built-in fixed-width binary cell index; a cell's `suid` is a Python tuple such as `('N', 3, 7)`. Two figures are reported instead, both computed here rather than assumed: an information-theoretic lower bound, $\lceil \log_2(\#\text{cells}) \rceil$ bits (the fewest bits that could possibly distinguish every cell at that resolution), and a naive fixed-width tuple encoding, 3 bits for the resolution-0 face id (6 faces) plus 4 bits per subsequent digit ($N_{\text{side}}^2 = 9$ children per cell), i.e. what a straightforward from-scratch binary encoding of the `suid` tuple costs.

Table 97.2: rHEALPix storage cost: information-theoretic lower bound and naive tuple-encoding bits per cell, and total naive storage. Source: `data/rhealpix_real_tradeoff_results.json`.

| res | #cells | info. bits/cell | naive bits/cell | total naive (MiB) |
|---|---|---|---|---|
| 0 | 6 | 2.585 | 3 | 0.0000019 |
| 1 | 54 | 5.755 | 7 | 0.000045 |
| 2 | 486 | 8.925 | 11 | 0.000637 |
| 3 | 4,374 | 12.095 | 15 | 0.00782 |
| 4 | 39,366 | 15.265 | 19 | 0.0892 |
| 5 | 354,294 | 18.435 | 23 | 0.971 |

The naive per-cell encoding is 3.17–4.57 bits above the information-theoretic minimum at every resolution tested, a gap that narrows in relative terms as resolution grows (16% over the minimum at resolution 5, against 76% at resolution 1): a fixed 4-bit field for a 9-way digit wastes $4 - \log_2 9 \approx 0.83$ bits per digit, a fixed overhead whose relative weight shrinks as more digits accumulate. This is the same qualitative shape as H3's and S2's storage costs — monotone growth, no saturation — at a smaller per-level factor ($9\times$ against H3's $7\times$ and S2's $4\times$, reflecting rHEALPix's $N_{\text{side}} = 3$ aperture).

## 97.5 What this resolves, and what remains

This chapter establishes a third, structurally distinct point in this volume's distortion-cost taxonomy. H3: distortion present and largest at the coarsest resolutions, falling and saturating quickly. S2: distortion absent at the coarsest resolutions, rising and saturating slowly. rHEALPix: area distortion absent *at every resolution, by construction*, while a different distortion axis (shape, IPQ) follows H3's fast-saturation shape rather than S2's slow one. No DGGS examined so far in this volume avoids growing storage cost; the exponent alone varies (7, 4, 9).

**To Complete / Verify**

This closes part of Section 11.5's item (i): rHEALPix has now been independently reproduced from raw cell geometry, alongside H3 and S2; DGGRID and OpenEAGGR have not. Left open: the resolution-correspondence mismatch between this chapter's naive enumeration and Kmoch & Vasilyev's cell counts (Remark 97.1), named but not diagnosed, unlike Chapter 14's S2 investigation, which at least tested two candidate mechanisms; rHEALPix operational timing (encode, hierarchy step, neighbour query, mirroring Chapters 94 and 96) has not been attempted — Chapter 98 has since done this, finding a third distinct query-cost shape on every operation timed, none matching H3's or S2's pattern exactly; and the coarsest-resolution IPQ approximation artifact (Section 97.3) has not been quantified against a non-approximating, exact-boundary perimeter calculation, which `rhealpixdggs` may or may not expose and was not investigated — Section 99.2 has since done this, finding the artifact real but only partial: refinement removes 69% of the excess above the theoretical ceiling, with a 0.03463 residual surviving convergence, tentatively attributed to a spherical- scale effect not present in flat-geometry isoperimetric reasoning.

Chapter 98

# Timing rHEALPix's Operations: A Third, Distinct Pattern

## 98.1 Scope: the operational-timing item left open by Chapter 97

Chapter 97's closing todobox left rHEALPix operational timing unattempted, mirroring Chapters 94 and 96's query-cost measurements for H3 and S2. This chapter closes that item, timing the `rhealpixdggs` analogues of the same three operations: point → cell encoding, one step up the hierarchy, and a same-cell neighbour query.

**Methodological Note**

**What was done, precisely, and its limits.** Three operations were timed at each resolution 0–5: `RHEALPixDGGS.cell_from_point(res, (lon, lat), plane=False)` (coordinate → cell encoding); reconstructing a cell's parent by slicing its own `suid` tuple and passing the result back through `RHEALPixDGGS.cell()` (see Remark 98.1 below for why this indirect route, rather than a native `parent()` call, is what is timed); and `Cell.neighbors(plane=False)` (the cell's edge-neighbours — always 4 of them for rHEALPix's quadrilateral tessellation, never 6–7 as for H3's hex grid or 8 as for S2's quadtree adjacency; the three counts are not commensurable and no claim is made that they should be).
A pilot run showed `neighbors(plane=False)` costs roughly 250–390 $\mu$s per call in this environment — on the order of 50–100× h3-py's `grid_disk` and 5–10× s2sphere's `get_all_neighbors` (Tables 94.1 and 96.1). Running Chapters 94 and 96's original protocol (2,000-call warm-up, then 21 batches of 20,000 calls, 422,000 calls per operation per resolution) at that cost would take on the order of an hour per resolution for the neighbour operation alone. The protocol here is scaled down accordingly, and the scaling-down is disclosed rather than left implicit: a discarded warm-up of 500 calls, then 11 timed batches of 2,000 calls each (23,500 calls per operation per resolution instead of 422,000), median per-call time reported, run twice independently. Parent and neighbour timings are measured on cells built *outside* the timed loop (an untimed pool of `cell_from_point` calls), so only the parent-reconstruction or neighbour-lookup step itself is timed, not encoding plus that step together. Same machine class as Chapters 94 and 96: single-core container (`nproc`=1), Intel Xeon @ 2.80GHz, Python 3.12.3, `rhealpixdggs` 0.6.0. Absolute timings are not portable beyond this environment; only the pattern surviving both runs is reported as a finding, and the smaller sample size relative to the H3/S2 chapters is an explicit limitation of what follows, not silently matched to their precision. Full code: `code/rhealpix_operational_timing.py`; full output (second of two runs, the one tabulated below): `data/rhealpix_operational_timing_results_run2.json` (first run: `…run1.json`).

*Remark* 98.1 (No native parent accessor). Unlike h3-py's `cell_to_parent` and s2sphere's `Cell.parent(level)`, `rhealpixdggs` 0.6.0 exposes no dedicated parent method on either `Cell`

or `RHEALPixDGGS`: neither class defines a function named `parent` anywhere in the installed package. The only documented route from a cell to its parent is to slice one element off its own `suid` tuple (e.g. `('N', 3, 7)` → `('N', 3)`) and pass the shortened tuple back through `RHEALPixDGGS.cell()`, which re-validates and reconstructs a full `Cell` object from it. What is timed below as "`parent`" is therefore this reconstruction, not a library-native traversal step, and the two are not doing the same amount of work: H3's and S2's parent calls look up an existing relationship in a fixed-width index, while rHEALPix's must rebuild a `Cell` object from scratch. This asymmetry is itself a finding about the three APIs' designs, not an artifact to be smoothed away by choosing a different rHEALPix call — there is no other call to choose.

## 98.2 The measured table

Table 98.1: Median per-call time (microseconds) for three rHEALPix operations across resolutions 0–5, second of two independent timing runs. Source: `code/rhealpix_operational_timing.py`, `data/rhealpix_operational_timing_results_run2.json`.

| res | `cell_from_point` | parent (rebuilt) | `neighbors` |
|---|---|---|---|
| 0 | 35.695 | — | 241.844 |
| 1 | 49.022 | 1.481 | 335.264 |
| 2 | 50.889 | 1.713 | 380.820 |
| 3 | 53.022 | 1.726 | 387.690 |
| 4 | 54.199 | 1.852 | 359.728 |
| 5 | 52.980 | 2.090 | 353.006 |

**Key Result**

None of the three operations reproduces another scheme's shape exactly, and all three differ from each other within this one chapter. `cell_from_point` jumps from resolution 0 to resolution 1 ($35.7 \rightarrow 49.0\,\mu$s, a 37% increase) and is then roughly flat with mild noise through resolution 5 ($49.0$–$54.2\,\mu$s, no further monotonic trend): a step-then-plateau shape with no counterpart in H3's smooth 10%-over-the-whole-range rise (Table 94.1) or S2's fully flat encoding cost (Table 96.1). The rebuilt `parent` step, small in absolute terms throughout, is neither flat like H3's `cell_to_parent` ($0.63$–$0.67\,\mu$s, no trend) nor S2's `parent` ($1.14$–$1.23\,\mu$s, no trend): it grows monotonically and by a wider relative margin, $1.481 \rightarrow 2.090\,\mu$s from resolution 1 to 5, a 41% increase confirmed in both independent runs (run 1: $1.503 \rightarrow 2.053\,\mu$s, the same shape). This is consistent with Remark 98.1's point that the operation is a `suid`-tuple slice-and-rebuild rather than an index lookup: a longer `suid` at higher resolution means more tuple elements to copy and re-validate on every call, a cost with no analogue in H3's or S2's fixed-width positional indices. `neighbors` rises from resolution 0 to resolution 1 ($241.8 \rightarrow 335.3\,\mu$s) and then plateaus, with noise but no further clear trend, through resolution 5 ($335.3$–$387.7\,\mu$s) — a third shape again: neither H3's fully flat `grid_disk` nor S2's falling-then-plateauing `get_all_neighbors` (Remark 96.1), but rising once, early, then plateauing.

*Remark* 98.2 (The neighbour-query cost gap is not a compiled-versus-interpreted story). `neighbors(plane=False)` costs 242–388 $\mu$s per call here, against S2's `get_all_neighbors` at $40.0$–$63.8\,\mu$s and H3's `grid_disk(k=1)` at $5.9$–$6.4\,\mu$s (comparing overall ranges across the resolutions each chapter tested, not resolution-matched, since the three schemes' resolution numbers do not denote comparable cell sizes) — roughly a $4$–$10\times$ gap against s2sphere and a 38–

66× gap against h3-py. The natural first guess, that this reflects compiled C code (h3-py) against interpreted Python (rhealpixdggs), does not by itself explain the s2sphere gap: inspection of the installed packages in this environment shows h3-py ships compiled Cython extension modules (`h3/_cy/*.so`) wrapping Uber's C library, while s2sphere ships only `.py` files with no compiled extension at all — s2sphere is pure Python, the same as `rhealpixdggs`, yet still 4–10× faster at the same kind of neighbour query. The gap between two pure-Python libraries must therefore come from what each one's neighbour-query code actually does per call, not from compiled-versus-interpreted execution. One concrete, visible difference: `rhealpixdggs`'s `Cell.neighbors()` branches on the cell's ellipsoidal shape (`quad`, `cap`, `skew_quad`, or `dart`), and for three of those four branches it computes each candidate neighbour's nucleus coordinates and sorts them by longitude to assign directional labels — work with no counterpart in a fixed hex-ring or quadtree-adjacency lookup. Whether this branching-and-sorting cost, as opposed to some other implementation difference not inspected here, is what actually accounts for the gap is not tested by timing the branches separately, and is left as a named but unconfirmed mechanism, in the same spirit as Remark 96.1's treatment of S2's own unexplained shape.

## 98.3 What this adds to the rHEALPix real trade-off

Chapter 97 showed that rHEALPix's area distortion is exactly zero by construction at every resolution, that its shape distortion (IPQ) saturates early like H3's, and that its storage cost grows without bound like both H3's and S2's, at a per-level factor of 9×. This chapter adds a fourth axis and finds no clean repeat of any earlier pattern on it: query cost is markedly higher in absolute terms than either H3 or S2 for the one operation (`neighbors`) that is directly comparable in kind across all three schemes, and follows a step-then-plateau or rise-then-plateau shape not seen in either earlier chapter. Taken together with Remark 94.1 and Chapter 96's finding that query cost does not track storage cost for H3 or S2 either, the pattern across all three schemes examined so far in this volume is that each of the four axes measured (area distortion, shape distortion, storage cost, query cost) behaves largely independently of the others and independently across schemes: no single scheme is uniformly cheapest or most expensive, and no single axis predicts another. For an application choosing among H3, S2, and rHEALPix, this chapter's evidence says query latency and index-refinement storage cost must be evaluated as separate, scheme-specific quantities rather than assumed to move together.

**To Complete / Verify**

This closes Chapter 97's operational-timing item. What remains open: the branching-and-sorting mechanism proposed in Remark 98.2 for the neighbour-query cost gap is named but not isolated by direct measurement of individual code paths — Section 99.1 has since done this, confirming the branch as the primary driver (the `quad` branch costs less than a single `nucleus()` call, the other three cost roughly 4× their own `nucleus()` cost) while surfacing a second, smaller, still-unisolated mechanism (Remark 99.1); whether DGGRID or OpenEAGGR would show a fourth distinct query-cost shape remains entirely untested by this volume; the smaller sample size used here relative to Chapters 94 and 96 (disclosed above) means the rHEALPix figures carry more measurement uncertainty than the H3 and S2 ones, not yet quantified with confidence intervals for any of the three schemes; and, as for H3 and S2, whether any of these patterns survive under concurrent load or a compiled-language binding to the same rHEALPix logic is not attempted here.

Chapter 99

# Two Open Items Closed: The Neighbour-Branch Mechanism and the Coarsest-Resolution IPQ Artifact

This chapter closes two items left open by the two preceding rHEALPix chapters: Remark 98.2's named-but-unisolated branching mechanism for the neighbour-query cost gap (Chapter 98), and Section 97.3's coarsest-resolution IPQ approximation artifact, left unquantified against a non-approximating boundary (Chapter 97). Both items are pursued here because both were within reach with the tools already installed, not because they are related to each other; they share only their rHEALPix subject matter.

## 99.1 Isolating the neighbour-branch mechanism

**Methodological Note**

**What was done, precisely, and its limits.** `Cell.neighbors(plane=False)` was timed separately for cell pools grouped by `ellipsoidal_shape` (`quad`, `cap`, `skew_quad`, `dart`), instead of pooled across whatever shapes a random resolution's point sample happened to produce, as Chapter 98 did. Because shape availability depends on resolution — resolution 0 has only `quad` and `cap`; `skew_quad` and `dart` first appear at resolution 1; `cap` is essentially exhausted by resolution 3 in this `N_side=3` layout, since the two polar caps occupy a fixed, resolution-independent handful of cells while quad-type cell count grows with resolution — each shape's pool was filled by sampling across resolutions 0–3 rather than one resolution at a time; the resolution mix used for each shape is disclosed in the JSON output and reported in Table 99.1's footnote below. This differs from Chapter 98's single-resolution-at-a-time protocol by design: this chapter's question is whether the branch predicts the cost, not how cost evolves with resolution, so pooling across resolutions to fill each shape bucket does not compromise the comparison, provided the mix is disclosed.
Alongside `neighbors(plane=False)` itself, `Cell. nucleus(plane=False)` was timed directly on the same cell pools. This is not a component extracted from inside `neighbors()` by patching the library — it is the same public method, called directly — so it lower-bounds, rather than exactly equals, the per-branch nucleus cost paid inside `neighbors()` (which also does a longitude sort and, for two of the four branches, a temporary prime-meridian shift, neither timed in isolation here).
Same batch protocol as Chapter 98: 500 discarded warm-up calls, then 11 timed batches of 2,000 calls each, median per-call time, run twice independently. Same machine class throughout this volume: single-core container, Python 3.12.3, `rhealpixdggs` 0.6.0. Full code: `code/rhealpix_neighbor_branch_timing.py`; full output (second of two runs, tabulated below): `data/rhealpix_neighbor_branch_timing_results_run2.json`.

Table 99.1: Median per-call time (microseconds) for `neighbors(plane=False)` and `nucleus(plane=False)`, timed separately per `ellipsoidal_shape`, second of two independent runs. Resolution mix per shape (cells drawn from resolutions 0–3): quad even across 0–3 (≈28k cells each); cap 80% res 0, 20% res 1, $<$ 1% res 2; skew_quad and dart drawn from res 1–3 only (absent at res 0). Source: `code/rhealpix_neighbor_branch_timing.py`, `data/rhealpix_neighbor_branch_timing_results_run2.json`.

| shape | `neighbors` | `nucleus` | 4×`nucleus` |
|---|---|---|---|
| quad | 19.628 | 64.948 | 259.79 |
| cap | 312.727 | 71.056 | 284.22 |
| skew_quad | 430.694 | 74.143 | 296.57 |
| dart | 412.079 | 76.131 | 304.52 |

**Key Result**

The branch predicts the cost. `quad` — the branch that Remark 98.2 identified as doing four dict lookups against already-computed planar neighbours, with no `nucleus()` call at all — costs 19.628 $\mu$s per `neighbors()` call, *cheaper than a single call to* `nucleus()` *alone* (64.948 $\mu$s): direct confirmation that the quad branch pays none of the nucleus cost, not merely a plausible inference from reading the source. The other three branches, which each call `nucleus()` on all four planar neighbours, cost an order of magnitude more: `cap` at 312.727 $\mu$s is within 28.5 $\mu$s (9%) of 4× its own measured `nucleus()` cost (284.22 $\mu$s), consistent with cap's branch doing exactly four nucleus calls plus a longitude sort and little else. `skew_quad` (430.694 $\mu$s) and `dart` (412.079 $\mu$s) exceed their own 4×`nucleus()` figures by a wider margin, 134.1 $\mu$s (45%) and 107.6 $\mu$s (35%) respectively — both runs agree on this ordering (run 1: quad 18.717 $\mu$s, cap 290.928 $\mu$s, skew_quad 420.094 $\mu$s, dart 426.299 $\mu$s; the same shape throughout). Remark 98.2's branching-and-sorting mechanism is therefore confirmed as the primary driver of the neighbour-query cost gap, isolated by direct measurement rather than left as a named-but-untested hypothesis.

*Remark* 99.1 (A second, smaller, still-open mechanism). Confirming the primary mechanism surfaces a secondary one that was not visible before the branches were separated: `skew_quad` and `dart` cost more than `cap` (430.7 and 412.1 $\mu$s against 312.7 $\mu$s) despite all three branches calling `nucleus()` the same number of times (four) and all three sorting four candidates. Inspecting the source (`rhealpixdggs/cell.py`, `Cell.neighbors()`) shows what `skew_quad` and `dart` do that `cap` does not: both temporarily reassign `self.rdggs.ellipsoid.lon_0` to move the prime meridian to the cell's own nucleus longitude before computing neighbour nuclei, then restore it afterward — two extra attribute writes per call, plus (for `skew_quad`) two `max`/two `min` calls with list removal in place of a single `sort` call for `cap`. `lon_0` is a plain instance attribute in this package, not a property with side effects (`rhealpixdggs/ellipsoids.py` sets it with a bare `self.lon_0 = lon_0` in `__init__`, and no setter elsewhere), so the reassignment itself should be $O(1)$ and cannot by itself explain a 100–134 $\mu$s gap. What remains unclear is whether shifting `lon_0` changes the cost of the four `nucleus()` calls that follow it (for instance if the underlying projection code takes a different branch near a shifted meridian), whether the `max`/`min`-with-removal pattern costs meaningfully more than `cap`'s single `sort` on a four-element list (implausible at this scale in ordinary Python, but not measured here), or some other difference in the `skew_quad` and `dart` branches not identified by this inspection. This second mechanism is, in the same spirit as the first, named but not yet isolated by direct measurement, and is left open.

## 99.2 Quantifying the resolution-0 IPQ artifact

**Methodological Note**

**What was done, precisely, and its limits.** Section 97.3's resolution-0 IPQ figures used a 4-vertex geodesic-chord perimeter (haversine distance summed over `Cell.vertices(plane=False)`). This section replaces those 4 chords with a refined boundary and re-measures IPQ. `Cell.boundary(n, plane=False)` could not be used directly for this: for `ellipsoidal_shape` in `{quad, cap}` — which is every one of the 6 resolution-0 cells, confirmed empirically here (4 `quad`, 2 `cap`), not assumed — the library's own `boundary()` short-circuits to `vertices()` regardless of `n`, on the documented assumption that 4 corners already represent those two shapes' ellipsoidal edges adequately. That assumption is exactly what this section tests, so it cannot supply the refinement tool.
Instead, this section builds its own refinement from a different public method the library does expose for every shape: `Cell.boundary(n, plane=True)` gives $n$ evenly spaced points per edge in *plane* coordinates, exact by construction since planar rHEALPix cells are true squares; each plane point is then mapped to a geographic coordinate with `RHEALPixDGGS.rhealpix(u, v, inverse=True, region=cell.region())` — the identical inverse-projection call the library's own `boundary()` uses internally for the `skew_quad/dart` case, applied here to `quad/cap` instead. No new geometric model is introduced; this is the library's own forward machinery run at finer sampling than `boundary()` chooses to run it for these two shapes.
One implementation pitfall is disclosed because it produced a first-draft result wrong by roughly two orders of magnitude before being caught: `RHEALPixDGGS.rhealpix(inverse=True)` returns degrees, not radians, when the underlying ellipsoid has `radians=False` — true of `WGS84_ELLIPSOID` as used throughout this volume (checked directly: `WGS84_ELLIPSOID.radians` is `False`) — despite the method's own docstring example using a different, radians-flagged ellipsoid instance (`UNIT_003_RADIANS`) and so showing radians output. A first version of this script applied `math.degrees()` to an already-degree value, inflating apparent longitudes to the tens of thousands and collapsing every IPQ toward zero; this was caught by comparing against the library's own `boundary(n=3, plane=False)` doctest output for a known cell, which is unambiguously in degrees, and fixed before any figure below was produced. The fix (removing the redundant conversion) is what produced every number reported here; no numbers from the buggy version appear anywhere in this monograph.
Convergence, not a single refined value, is reported: perimeter and IPQ were recomputed at $n \in \{2, 4, 8, 16, 32, 64, 128\}$ (4 to 508 boundary points per cell) so that the refined figure is evidenced as stable rather than asserted. Radius convention: `WGS84_ELLIPSOID.R_A` (authalic radius) throughout, matching the area convention already used in Chapter 97, applied identically to both the naive ($n = 2$) and refined perimeters so only the boundary sampling differs between them. Full code: `code/rhealpix_ipq_artifact_refinement.py`; full output: `data/rhealpix_ipq_artifact_refinement_results.json`.

**Key Result**

Refinement confirms an approximation artifact, but only a partial one. The $n = 2$ mean, 1.11008, matches Chapter 97's reported 1.10911 closely enough (difference 0.00097, attributable to minor floating-point and radius-handling differences between the two independently written scripts) to confirm this section's method reproduces the original figure before refining it. Refinement then converges cleanly: the mean falls from 1.11008 ($n = 2$)

Table 99.2: Resolution-0 IPQ (mean, std across the 6 cells) as a function of boundary refinement $n$ (points per edge; $4n - 4$ points per cell). $n = 2$ reproduces Chapter 97's original 4-vertex figure. Source: `code/rhealpix_ipq_artifact_refinement.py`.

| $n$ | pts/cell | IPQ mean | IPQ std |
|---|---|---|---|
| 2 | 4 | 1.11008 | 0.16290 |
| 4 | 12 | 1.04173 | 0.12419 |
| 8 | 28 | 1.03591 | 0.12103 |
| 16 | 60 | 1.03490 | 0.12048 |
| 32 | 124 | 1.03469 | 0.12037 |
| 64 | 252 | 1.03464 | 0.12034 |
| 128 | 508 | 1.03463 | 0.12033 |

to 1.03463 ($n = 128$), with the last three steps ($n = 32, 64, 128$: $1.03469 \to 1.03464 \to 1.03463$) changing by 0.00006 total, evidence of convergence rather than an arbitrary stopping point. Of the original 0.11008 excess above the theoretical flat-geometry ceiling of 1, refinement removes 0.07545 of it (69%) — the coarse 4-chord approximation *is* responsible for most of the apparent super-circularity flagged in Chapter 97. But 0.03463 of the excess survives convergence and does not go to zero: the refined, 508-point-per-cell boundary still gives a mean IPQ above 1, most visibly for the two `cap` cells ($1.34044 \to 1.20481$ from $n = 2$ to $n = 128$, still 20% above the flat ceiling after refinement) against the four `quad` cells ($0.99489 \to 0.94954$, which cross *below* the ceiling once refined).

*Remark* 99.2 (Why a genuine, converged IPQ above 1 is not itself surprising). $4\pi A/P^2 \leq 1$ is a planar (Euclidean) isoperimetric fact: among regions of a fixed perimeter, the circle maximizes area, and no planar figure exceeds the ratio a circle achieves. It is not automatically a spherical fact. On a sphere, for a fixed *geodesic* perimeter, the area-maximizing shape is a spherical cap, and a spherical cap's area grows faster relative to its perimeter than a flat disc's does as the cap grows larger, because the cap's boundary circle has a smaller geodesic radius on the sphere than the corresponding flat disc would need to enclose the same area (a spherical-excess effect, not examined elsewhere in this volume at a comparable angular scale). At small angular scales this effect is negligible and $4\pi A/P^2 \leq 1$ holds to good approximation, consistent with resolutions 1–5's converged means safely under 1 (Table 97.1). At resolution 0, where each cell covers a full 60–90° of angular extent (a hemisphere quarter for the equatorial `quad` cells, a full polar cap for `cap`), the effect is no longer negligible. This is offered as a plausible geometric account of the residual 0.03463, consistent with the `cap` cells (the more compact, more nearly circular shape, and so closer to the shape that maximizes the spherical effect) showing the larger residual excess of the two shapes present at resolution 0 — but it is reasoning from general spherical geometry, not a claim independently verified by a purpose-built check in this volume (e.g. by directly computing the IPQ of an exact spherical cap of matching area on the same ellipsoid and confirming it, too, exceeds 1 by a comparable margin), and is reported at that lower confidence level.

**To Complete / Verify**

This closes part of Chapter 98's neighbour-branch-mechanism item and part of Chapter 97's IPQ-artifact-quantification item. What remains open: Remark 99.1's secondary mechanism (why `skew_quad/dart` cost 100–134 $\mu$s more than 4×their own measured `nucleus()` cost, more than `cap` does) is named but not isolated, exactly the position Remark 98.2 was in before this chapter, one level down; whether DGGRID or OpenEAGGR would show a fourth distinct query-cost shape remains entirely untested by this volume,

as before; Remark 99.2's spherical-excess account of the residual $0.03463$ IPQ gap is reasoned, not independently verified by a dedicated exact-spherical-cap check; and this chapter's IPQ refinement was applied only at resolution 0, where the artifact was originally flagged — whether the much smaller $n = 2$-versus-refined gap implied by resolutions 1–5's already-converged, sub-1 IPQ means (Table 97.1) would show the same partial-artifact pattern at smaller magnitude, or none at all, was not checked at any resolution above 0.

Chapter 100

# The Spherical Ceiling Checked: A Latitude-Convention Artifact, Not (Only) Spherical Excess

Remark 99.2 closed the previous chapter with a named but unverified account: resolution 0's residual $0.03463$ mean IPQ excess above the flat-geometry ceiling of $1$ was attributed, reasoning from general spherical geometry, to spherical excess, and flagged explicitly as not confirmed by "directly computing the IPQ of an exact spherical cap of matching area ... and confirming it, too, exceeds $1$ by a comparable margin." This chapter runs exactly that check. It does not simply confirm the account: it first appears to *break* it, then, once the reason why is traced, ends up confirming it far more precisely than a mere "comparable margin" would have — while surfacing an unrelated, previously invisible measurement bias in Chapter 99's own refined perimeters.

## 100.1 An exact ceiling, and an apparent violation of it

A spherical cap of angular radius $\theta$ on a sphere of radius $R$ has area $A = 2\pi R^2(1-\cos\theta)$ and a boundary of physical length $P = 2\pi R\sin\theta$, giving

$$\mathrm{IPQ}_{\mathrm{cap}}(\theta) = \frac{4\pi A}{P^2} = \frac{2(1-\cos\theta)}{\sin^2\theta} = \frac{2}{1+\cos\theta},$$

a closed form derived directly in `code/rhealpix_spherical_ipq_ceiling.py` (elementary spherical geometry, not drawn from an external source). By the spherical isoperimetric inequality — among regions of fixed area on a sphere, the cap uniquely minimizes perimeter, hence uniquely maximizes $4\pi A/P^2$ — this is not just *a* cap's IPQ, it is an exact upper bound on the IPQ of *any* region of that area on that sphere. Feeding each resolution-0 cell's own area (from Chapter 99's `data/rhealpix_ipq_artifact_refinement_results.json`, not recomputed independently, so the ceiling is checked against the exact area figure already in use rather than a fresh figure that could silently drift from it) into this formula gives a single shared ceiling for all six resolution-0 cells, $\theta = 48.190°$, $\mathrm{IPQ}_{\mathrm{ceiling}} = 1.20000$ (all six share one area under rHEALPix's equal-area construction, confirmed directly from the source JSON rather than assumed).

Run against Chapter 99's own refined ($n = 128$) figures, this ceiling is comfortably respected by the four `quad` cells ($0.94954$, a margin of $+0.250$) but — unlike every other check run so far in this volume, which is precisely what made it worth reporting rather than quietly redone — appears to be *violated* by both `cap` cells: $1.20481$ against a ceiling of $1.20000$, a margin of $-0.00481$. Because the isoperimetric inequality is exact, a stably reproducible violation of it is not a possible outcome for two internally consistent quantities compared correctly; extending Chapter 99's own $n$-ladder for a `cap` cell well past $n = 128$, to $n = 4096$ (508 boundary points $\rightarrow$ over $16{,}000$), shows the naive IPQ converging cleanly to $1.204799$, not receding toward the

ceiling as $n$ grows — so this is not a lingering chord-approximation gap either. Something about the comparison itself, not the geometry, had to be inconsistent.

## 100.2 Tracing the mechanism

**Methodological Note**

**What was done, precisely, and its limits.** The `rhealpixdggs` source was read directly (not assumed from its public docstrings) to find where geodetic and authalic latitude might be conflated: `pj_healpix.healpix_ellipsoid_inverse` calls `RHEALPixDGGS.rhealpix(…, inverse=True)`, which internally inverts the HEALPix sphere projection to an authalic latitude $\beta$ and then converts $\beta$ back to common (geodetic) latitude $\varphi$ via `auth_lat(…, inverse=True)` before returning — because a caller wanting ellipsoidal geographic coordinates naturally wants $\varphi$, not $\beta$. Chapter 99's own `refined_boundary_geo_deg` takes exactly this returned $\varphi$ and feeds it straight into a spherical haversine formula on a sphere of radius $R_A$ — but the equal-area construction underlying `cell.area(plane=False)` (and hence this chapter's ceiling, which is computed from that same area) lives on the *authalic* sphere, whose latitude coordinate is $\beta$, not $\varphi$. The two are not the same number: for WGS84, $|\beta-\varphi|$ reaches a maximum of about $0.1283°$ near $\varphi = 45.05°$, and is about $0.1275°$ at resolution 0's quad/cap boundary latitude of $41.8103°$ — not a negligible gap against an IPQ effect measured in tenths of a percent. This is a latitude-*convention* bug in the comparison, not a defect in `rhealpixdggs` itself, which returns exactly the geodetic coordinate its own documentation promises; the mismatch was introduced by Chapter 99's haversine call treating that correctly-returned geodetic value as though it were the sphere's own angular coordinate.

`code/rhealpix_ipq_authalic_correction.py` repeats Chapter 99's exact boundary sampling (same `Cell.boundary(n, plane=True)` points, same inverse projection) but converts each returned $\varphi$ back to authalic $\beta$ with the library's own `auth_lat()` before the haversine call, so the perimeter is finally expressed on the same sphere as the area. Two checks confirm this is the right fix rather than a second coincidence: first, sampling a `cap` cell's boundary at $n = 8$ and inspecting the authalic latitude at every one of its 28 points gives a spread of $7\times10^{-15}$ degrees — zero to floating-point precision, i.e. the `cap` cell's boundary is an *exact* small circle of constant authalic latitude on the authalic sphere, exactly the shape the "cap" name promises and exactly the shape for which the isoperimetric ceiling is attained with equality, not just respected with margin. Second, re-running the $n$-ladder with this correction shows the `cap` IPQ converging monotonically *up to* the ceiling from below — $1.33448$ $(n = 2) \to 1.20005$ $(n = 16)$ $\to 1.200001$ $(n = 1024)$, overshoot-free, the residual shrinking by very close to $4\times$ per doubling of $n$ (consistent with an ordinary $O(1/n^2)$ chord-length approximation, the same kind Chapter 99 already characterized, now converging toward the true ceiling instead of past it).

**Key Result**

The apparent ceiling violation is fully explained, and the explanation strengthens Remark 99.2 rather than undermining it. Per-cell, at $n = 128$: the two `cap` cells' corrected IPQ is $1.20001$ against a ceiling of $1.20000$ (margin $-0.000007$, i.e. equality up to the same residual chord-approximation gap the $n$-ladder above shows shrinking toward zero) — `cap` cells are not merely *consistent with* the spherical-excess account, as Remark 99.2 could only claim before this check; they are, empirically, exact isoperimetric-optimal spherical caps under rHEALPix's own authalic construction, so all of their excess above $1$ is spher-

Table 100.1: Resolution-0 mean IPQ across the 6 cells, naive (geodetic-latitude, Chapter 99's original figures) versus authalic-latitude-corrected, by boundary refinement $n$. Source: `code/rhealpix_ipq_authalic_correction.py`.

| $n$ | IPQ mean (naive) | IPQ mean (corrected) | difference |
|---|---|---|---|
| 2 | 1.11008 | 1.10911 | +0.00097 |
| 4 | 1.04173 | 1.04111 | +0.00062 |
| 8 | 1.03591 | 1.03532 | +0.00059 |
| 16 | 1.03490 | 1.03432 | +0.00058 |
| 32 | 1.03469 | 1.03411 | +0.00058 |
| 64 | 1.03464 | 1.03406 | +0.00058 |
| 128 | 1.03463 | 1.03405 | +0.00058 |

ical excess, with no unexplained residual. The four `quad` cells move from 0.94954 (naive) to 0.95107 (corrected) — comfortably below the ceiling either way, so their qualitative status (sub-1, non-optimal shape) is unchanged; the +0.00058 population-mean correction (visible in Table 100.1's last column, stable across $n$) is the net of a +0.0048 `cap` correction and a partially offsetting −0.0015 `quad` correction, weighted 2-caps-to-4-quads.

## 100.3 Does the same bias appear at finer resolutions?

Chapter 99's own closing todobox flagged a related gap: its refinement "was applied only at resolution 0 ...whether ...resolutions 1–5 ...would show the same partial-artifact pattern at smaller magnitude, or none at all, was not checked at any resolution above 0." That todobox was about the $n$=2-versus-refined chord gap specifically, not this latitude- convention bias, but the question generalizes naturally to it, and `code/rhealpix_ipq_authalic_multires_check.py` answers it directly: for resolutions 0, 1 and 2, every cell's boundary was sampled at a fixed $n = 16$ (a check of order of magnitude, not a full convergence ladder), naive and corrected IPQ compared per cell, and both the population mean and the per-`ellipsoidal_shape` mean difference recorded.

Table 100.2: Naive-minus-corrected IPQ, mean by shape and population mean, at $n = 16$, resolutions 0–2. Source: `code/rhealpix_ipq_authalic_multires_check.py`.

| res | cells | pop. mean diff | `cap` | `quad` | `skew_quad` | `dart` |
|---|---|---|---|---|---|---|
| 0 | 6 | +0.00058 | +0.00480 | −0.00153 | — | — |
| 1 | 54 | −0.000004 | +0.00851 | −0.00191 | +0.00349 | +0.00296 |
| 2 | 486 | −0.00012 | +0.00895 | −0.00192 | +0.00331 | +0.00385 |

**Key Result**

The naive intuition — that a latitude-convention bias tied to a cell's angular extent should shrink as cells get smaller with finer resolution — is wrong, and worth stating precisely because it is wrong. Table 100.2's population-mean column does shrink sharply (from +0.00058 at resolution 0 to −0.000004 at resolution 1), but the per-shape columns do not: the `cap` bias actually *grows slightly* with resolution (+0.0048 → +0.0085 → +0.0090) and `quad`'s stays essentially flat (−0.0015 → −0.0019 → −0.0019), with the newly-appearing `skew_quad` and `dart` shapes showing their own comparable-magnitude, resolution-stable

biases from the moment they first exist at resolution 1. The population mean's collapse is a composition effect, not a resolution effect: it is the near-cancellation of a persistently positive `cap/skew_quad/dart` bias against a persistently negative `quad` bias, weighted by each shape's share of the cell population, which happens to come out small at resolutions 1–2 under this `N_side=3` layout. The underlying mechanism (local difference between the geodetic-latitude scale factor and the authalic one) is a function of a cell's *position* on the ellipsoid, not its *size*, so there is no reason to expect it to vanish with finer resolution, and this check confirms it does not. This means Chapter 99 and Chapter 97's reported *population-mean* IPQ figures at resolutions 1–5 are likely close to what an authalic-corrected recomputation would give (per this same near-cancellation), but any per-shape or per-cell breakdown at those resolutions, had one been reported, would carry the same order-of- magnitude bias this section quantifies at resolution 0 — none was reported there, so nothing already published in this volume needs correction, but none should be added later without this correction applied.

**To Complete / Verify**

This closes Remark 99.2's open verification item: the spherical-excess account is now not merely plausible but exact for `cap` cells, up to a chord-approximation residual already shown shrinking toward zero. It also closes part of Chapter 99's closing todobox, the resolutions-1–5 question, for the latitude-convention bias specifically (though not for the original $n$=2-versus-refined chord-gap question that todobox actually asked, which remains unchecked above resolution 0). What remains open: this section's $n = 16$ multi-resolution check is a single-refinement snapshot, not a convergence ladder, at resolutions 1 and 2 — whether the reported per-shape means themselves depend materially on $n$ was not tested there, only at resolution 0 (Table 100.1); resolutions 3–5 were not checked at all, on the (untested) assumption that a bias already shown resolution-independent through resolution 2 continues to be; and whether the near-cancellation behind resolutions 1–2's small population-mean bias is a structural property of `N_side=3`'s specific shape-population ratios or a coincidence of this particular grid parameterization was not investigated and would require repeating this check under a different `N_side`.

Chapter 101

# The Latitude-Convention Bias, Checked for Stability in $n$ and Extended to Resolution 3

Chapter 100's closing todobox left three items open. This short chapter closes two of them: whether the per-shape geodetic/authalic bias reported at a single $n = 16$ (Table 100.2) was itself converged in $n$, and whether the bias's resolution-independence, shown through resolution 2, continues at resolution 3. The third item (whether the population-mean near-cancellation is structural to `N_side=3` or coincidental, which would require a different `N_side`) is not attempted here and remains open; this chapter is scoped to what could be checked with the same grid parameterization already in use throughout this volume's rHEALPix chapters.

**Methodological Note**

**What was done, precisely, and its limits.** `code/rhealpix_ipq_authalic_res1ladder_res3.py` runs two checks. First, a genuine $n$-ladder ($n = 2, 4, 8, 16, 32, 64$) at resolution 1, per `ellipsoidal_shape`, using the same naive-vs-authalic- corrected IPQ comparison as Chapter 100. Second, the same fixed-$n = 16$ snapshot Chapter 100's multires check used, extended to resolution 3 (4374 cells). Resolution 4 (39,366 cells) and resolution 5 (354,294 cells) were not run: at this script's own measured cost of roughly 8.7 ms per cell for the naive-plus-corrected pair at $n = 16$, resolution 4 would take on the order of 6 minutes and resolution 5 on the order of 50 minutes, disproportionate to a single follow-up check, and are left open rather than attempted under time pressure or run in the background unverified.

Table 101.1: Naive-minus-corrected IPQ, mean by shape, resolution 1, as a function of boundary refinement $n$. Source: `code/rhealpix_ipq_authalic_res1ladder_res3.py`.

| $n$ | cap | quad | skew_quad | dart |
|---|---|---|---|---|
| 2 | +0.010485 | −0.001908 | +0.003607 | +0.002983 |
| 4 | +0.008695 | −0.001910 | +0.003500 | +0.002959 |
| 8 | +0.008534 | −0.001910 | +0.003488 | +0.002957 |
| 16 | +0.008506 | −0.001910 | +0.003486 | +0.002956 |
| 32 | +0.008500 | −0.001910 | +0.003486 | +0.002956 |
| 64 | +0.008498 | −0.001910 | +0.003486 | +0.002956 |

**Key Result**

Both items close cleanly, in the direction Chapter 100's reasoning anticipated rather than against it. Table 101.1 shows the $n = 16$ snapshot Chapter 100 used was already converged for three of the four shapes by $n = 8$ and effectively converged for all four by $n = 16$ itself (`cap`, the slowest to settle, moves only 0.000008 between $n = 32$ and

$n = 64$) — the same fast, chord-length- driven convergence Chapter 100's resolution-0 ladder already established, now confirmed at resolution 1 too, so the earlier snapshot's per-shape figures were not artifacts of an under-refined boundary. At resolution 3, the same four biases persist at closely comparable magnitude to resolutions 0–2 (`cap`$= +0.00900$, `quad`$= -0.00192$, `skew_quad`$= +0.00341$, `dart`$= +0.00391$, against resolution 2's $+0.00895$, $-0.00192$, $+0.00331$, $+0.00385$) — each shape's bias has now been shown stable across four resolutions spanning a 729-fold range in cell count (6 to 4374 cells), which is considerably stronger evidence for the "position, not size" account Chapter 100 offered than the original two-resolution check alone. The population mean at resolution 3, $-0.000128$, again small and again attributable to the same `quad`-versus-everything-else near-cancellation, this time with `quad` cells making up 2916 of 4374 cells (66.7%, against `skew_quad`'s 1352, `dart`'s 104, and `cap`'s fixed 2) — the shape-count ratios themselves shift with resolution even though the per-shape biases do not, so the smallness of the population mean at any given resolution remains, as Chapter 100 already flagged, a fact about that resolution's shape composition rather than a sign the underlying per-cell bias is shrinking.

**To Complete / Verify**

Two of Chapter 100's three open items are closed: the $n = 16$ snapshot is confirmed converged, and the resolution-independence of the per-shape bias is confirmed through resolution 3. What remains open: resolutions 4 and 5 were not run, for the compute-cost reasons disclosed above, so the "position, not size" account is confirmed only up to resolution 3, not through the full resolution 0–5 range this volume otherwise reports; and Chapter 100's third item, whether the population-mean near-cancellation is a structural property of `N_side=3`'s shape-count ratios specifically, remains completely untested and would require rebuilding the grid under a different `N_side`, which no chapter in this volume has done so far — every rHEALPix result reported to this point, across Chapters 97 to 100 and this chapter, uses `N_side=3` throughout, so none of it speaks to whether these findings are specific to that choice.

Chapter 102

# S2's Discrete Shape Cost: A Cleaner Story Than rHEALPix's

## 102.1 Scope: the second grid for the three-way comparison

Chapter 103's closing todobox named the gap directly: rHEALPix has all three distortion measures — continuous Tissot (Chapter 31), finite ring-placement, and discrete IPQ (Chapter 99) — placed side by side; S2 has only the first. This chapter supplies the discrete IPQ side for S2, using the identical refined-boundary protocol Chapter 99 built for rHEALPix, so the two grids' shape-cost curves are, for the first time in this volume, produced by the same method rather than compared across different ones. It does not attempt the ring-placement side (Section 102.5).

**Methodological Note**

**What was done, precisely, and its limits.** For every S2 cell at levels 0–6 (native enumeration, `CellId.begin(level)` stepped to `CellId.end(level)`, the same method Chapter 95 used), a refined perimeter is built the same way Chapter 99 refined rHEALPix's: not from the 4 corner vertices alone, but from $n$ points per edge, using the library's own coordinate machinery at finer sampling than any single chapter's original method chose to run it at. Concretely: each cell's 4 corners (`Cell.get_vertex(k)`) are converted to this cell's own face's $(u, v)$ via the module-level `face_xyz_to_uv(face, point)` — forcing the cell's own face rather than letting the library auto-select one, the same discipline Chapter 32 used and for the same reason (auto-selection is exactly what produced that chapter's artifact) — then to $(s, t)$ via `CellId.uv_to_st`. $n$ points are then linearly interpolated along each edge *in* $(s, t)$ *space* (the grid's own uniform coordinate, so a straight interpolation there is the correct analogue of rHEALPix's `Cell.boundary(n, plane=True)`), each mapped back through `CellId.st_to_uv` and `face_uv_to_xyz` to a unit vector, and the refined perimeter is the sum of consecutive-point geodesic angles (`Point.angle()`), the same chord method Chapter 95 already used for the unrefined case. Area is `Cell.exact_area()` throughout, unchanged from Chapter 95; everything is on the unit sphere ($R = 1$), also matching that chapter's convention, and IPQ is scale-invariant so this choice does not affect the ratio. $n = 1$ (one point per edge, the corner itself) exactly reproduces Chapter 95's original 4-vertex perimeter; $n = 2, 4, 8, \ldots$ refine further. Full code: `code/s2_ipq_refined.py`; full output: `data/s2_ipq_refined_results.json`.

Table 102.1: Naive ($n = 1$, 4 vertices) versus refined ($n = 32$, 128 points/cell) IPQ mean, levels 0–6. Source: `code/s2_ipq_refined.py`, `data/s2_ipq_refined_results.json`.

| level | #cells | IPQ mean (naive, $n = 1$) | IPQ mean (refined, $n = 32$) |
|---|---|---|---|
| 0 | 6 | 1.085578 | 1.085578 |
| 1 | 24 | 0.838201 | 0.838201 |
| 2 | 96 | 0.785382 | 0.785382 |
| 3 | 384 | 0.771778 | 0.771778 |
| 4 | 1,536 | 0.768376 | 0.768376 |
| 5 | 6,144 | 0.767526 | 0.767526 |
| 6 | 24,576 | 0.767314 | 0.767314 |

## 102.2 The refinement check: no artifact to find, anywhere

**Key Result**

Refinement changes nothing, at any level tested: the two columns agree to at least six significant figures everywhere, with the actual per-cell difference on the order of $10^{-15}$ (floating-point noise, confirmed by inspecting `data/s2_ipq_refined_results.json` directly rather than the six-figure table above). This is the opposite finding from Chapter 99's rHEALPix result, where refinement moved the resolution-0 mean from 1.11008 to 1.03463, a genuine 6.8% change. For S2, there is no refinement artifact at resolution 0, or at any of the other six levels checked.

This null result is not merely observed; it follows from the same construction already used to explain Chapter 32's corner reversal, and is checked directly rather than left as a plausibility argument. Along any one edge of an S2 cell at any level, exactly one of the two quadratic coordinates, $s$ or $t$, is *held fixed* (that is what makes it a cell edge in a grid uniform in $(s, t)$). Because $S2$'s quadratic correction acts coordinatewise — $u = \text{st_to_uv}(s)$ depends on $s$ alone, not $t$ — holding $s$ fixed along an edge also holds $u$ fixed along that edge, whatever $t$ (and hence $v$) does. An edge of constant $u$ is, in the gnomonic $(u, v)$ plane, a straight line – and gnomonic (central) projection has the defining property that *every* straight line in the tangent plane is the image of a great circle on the sphere, not only lines through the plane's origin. So every S2 cell edge, at every level, is an exact great-circle arc, and a single chord between its two corners already *is* the geodesic distance along it; no amount of refinement can shorten or lengthen what is already exact. This was checked directly, not only inferred: for a representative level-3 edge, every interior point sampled at $n = 8$ was tested against the great-circle plane defined by that edge's own two corners ($\hat{n} = \hat{c}_0 \times \hat{c}_1$, normalized), and every point's deviation from that plane ($|\hat{p} \cdot \hat{n}|$) was on the order of $10^{-17}$ across all seven interior points checked – the same floating-point-noise level as the perimeter comparison above, not a small but nonzero curvature signal.

Table 102.2: S2, levels 0–6: cell count and IPQ statistics (refined, $n = 32$, though Section 102.2 shows this equals the $n = 1$ figure at every level). Source: `code/s2_ipq_refined.py`, `data/s2_ipq_refined_results.json`.

| level | #cells | IPQ mean | IPQ std | IPQ min | IPQ max |
|---|---|---|---|---|---|
| 0 | 6 | 1.08558 | 0.00000 | 1.08558 | 1.08558 |
| 1 | 24 | 0.83820 | 0.00000 | 0.83820 | 0.83820 |
| 2 | 96 | 0.78538 | 0.01517 | 0.76135 | 0.80340 |
| 3 | 384 | 0.77178 | 0.01708 | 0.72024 | 0.78977 |
| 4 | 1,536 | 0.76838 | 0.01772 | 0.69974 | 0.78644 |
| 5 | 6,144 | 0.76753 | 0.01789 | 0.68980 | 0.78566 |
| 6 | 24,576 | 0.76731 | 0.01794 | 0.68494 | 0.78546 |

## 102.3 The genuine curve: S2's discrete shape cost

**Key Result**

S2's IPQ curve follows the same qualitative shape as rHEALPix's and H3's (Table 97.1): an H3-like, not an S2-$\sigma$-like, curve – sharp early movement that saturates, not a slow rise. The mean falls from $1.08558$ (level 0) to $0.83820$ (level 1) to $0.78538$ (level 2) and is essentially flat from level 3 on ($0.77178 \to 0.76838 \to 0.76753 \to 0.76731$, moving by only $0.00022$ across the last two levels and a $4\times$ cell-count increase). This is the opposite shape from this same grid's own $\sigma$(norm. area) curve (Table 95.1), which *rises* from exactly $0$ at levels 0–1 before saturating: S2's two distortion axes, area and shape, behave oppositely at the coarsest levels – cells start perfectly area-uniform but shape-distorted, then both curves settle into their respective plateaus by level 3–5. The level-0 mean above $1$ ($1.08558$, for genuinely exact great-circle edges, not an approximation) is, unlike rHEALPix's resolution-0 figure, not an artifact at all: it is a real property of the $6$ faces themselves – each face covers a quarter of a hemisphere, large enough for the same spherical-excess effect Remark 99.2 described for rHEALPix's polar caps to make a genuinely circular-ish large geodesic quadrilateral exceed the flat-geometry ceiling of $1$, now confirmed exactly rather than approximately since no chord shortening is involved.

## 102.4 Placing S2 in the three-way comparison

Table 102.3: Two measures for S2, by region: continuous Tissot $\omega$ (Table 31.2, corrected by Chapter 32 where the original table's last two entries were a face-crossing artifact) and discrete IPQ (Table 102.2, plus the level-4 argmin cell located directly).

| Region | Tissot $\omega$ | IPQ |
|---|---|---|
| Face center ($\lambda = \phi = 0°$) | minimum, $0°$ (exactly isotropic) | — |
| Face interior, resolution-0 (whole face) | — | $1.08558$ (above flat ceiling, real) |
| Approaching face boundary ($\theta \to \theta_c \approx 38.17°$) | rising smoothly, no interior max | — |
| Near cube corners, finer resolutions | — | minimum (least circular) cells located here |

**Key Result**

The two measures agree on where S2 is worst, and the agreement was checked by direct location, not by qualitative impression alone: the level-4 cell with the single lowest IPQ across all $1{,}536$ cells at that level ($0.69974$, well below the level's own mean of $0.76838$) is centered at $37.61°$N, $-45°$ longitude – within $2.3°$ of $35.26°$, the exact latitude of a cube corner ($\arcsin(1/\sqrt{3})$), at a longitude ($-45° \equiv 315°$) that is exactly one of the four corner longitudes on that latitude band. This is the discrete analogue of Chapter 32's continuous finding that $\omega$ rises monotonically toward the cube corner within each face, with no interior relief: both measures, produced by unrelated methods (one a continuous finite-difference Jacobian, the other a discrete refined-perimeter search over a whole resolution's cell population), independently point to the immediate neighborhood of the cube corners as S2's worst-behaved region, at both the local and the finite-cell scale. This is the same qualitative agreement Chapter 103 found for rHEALPix's belt/cap boundary, now established for S2's second and structurally different worst-case geometry (a vertex singularity rather than a one-dimensional seam).

## 102.5 What remains open

**To Complete / Verify**

This chapter closes the IPQ side of Chapter 103's S2 gap but not the ring-placement side: Chapters 62 and 63's finite-scale ring-placement ratio (outermost-versus-second-outermost latitude band boundary, against an idealized graticule) was not attempted here for S2 and is not addressed by this chapter. rHEALPix, DGGRID, and OpenEAGGR's own IPQ curves aside from rHEALPix's (already done) remain unmeasured by this method; only S2 and rHEALPix now have both a continuous Tissot field and a discrete, refinement-checked IPQ curve. The level-4 argmin-IPQ cell's proximity to a cube corner ($2.3°$ of latitude, exact longitude match) is reported as located, not as a general theorem that the argmin cell at every level sits this close; only level 4 was checked this way.

**Update:** The ring-placement side is now closed. Chapter 104 reruns Chapter 63's identical method at levels $0$–$3$, so S2 now has ring-placement, IPQ, and Tissot data across the same level range as rHEALPix. DGGRID's and OpenEAGGR's own IPQ curves remain the one item genuinely still open here: no chapter in this volume has yet built a resolution-swept IPQ curve for either grid by this method, and doing so is not attempted here.

**Update:** This item remains open for the same structural reason as Section 82.3's two items: a resolution-swept IPQ curve for DGGRID or OpenEAGGR requires the compiled DGGRID binary and the `eaggr` bindings to enumerate cells across several resolutions each, which this volume's software environment does not provide. Unlike a single missing figure, this is a multi-resolution sweep, so the toolchain cost would need to be paid and then re-paid across several enumeration runs before a single curve could be drawn; it is left explicitly open rather than attempted partially with one or two resolutions standing in for a full curve.

Chapter 103

# Local and Finite-Scale Distortion, Compared

## 103.1 Scope: three measures of the same grid, never placed side by side

This volume now has, for rHEALPix specifically, three genuinely different measurements of the same underlying construction, produced by three different methods at three different scales, none of them built with the others in mind: Chapter 31's continuous, point-wise Tissot indicatrix ($\omega$, computed from the library's own forward transform by finite differences); Chapters 62 and 63's finite-scale ring-placement ratio (how closely the real boundary between the outermost and second-outermost latitude bands matches an idealized graticule's prediction, at a specific resolution); and Chapter 99's discrete isoperimetric quotient (how compact realized cells are, averaged over whichever cells exist at a given resolution). This chapter places the three side by side for the first time and asks the obvious question: do they agree on where the grid is worst?

**Methodological Note**

**What this comparison can and cannot show.** The three quantities are not the same mathematical object. $\omega$ is a local, continuous property of a single point, independent of any resolution or cell boundary. The ring-placement ratio and the IPQ are both discrete, cell-based quantities, averaged or evaluated over whichever cells a specific resolution happens to produce, and both depend on resolution in ways $\omega$ by construction does not. No functional relationship between them is derived or fitted here; what follows is a qualitative comparison of *where*, in latitude, each measure is largest, not a claim that one predicts the other's numerical value.

## 103.2 rHEALPix: three measures, one qualitative story

**Key Result**

All three measures agree, qualitatively, that the polar cap and the belt/cap boundary are where rHEALPix is least well-behaved, and that the equatorial belt's interior (away from the boundary) is comparatively well-behaved: the Tissot $\omega$ minimum sits at $\phi \approx 20$–$25°$, squarely inside the belt and away from both the equator and the boundary; the resolution-0 quad (belt) cells' refined IPQ, $0.94954$, sits *closer* to the flat-geometry ceiling of $1$ than the cap cells' $1.20481$ does, in the same direction (cap more anomalous than quad); and the ring-placement ratio's largest deviation from $1$ across all three resolutions tested, $+26.6\%$ at resolution 2, occurs at the coarsest resolution, where the outermost ring sits nearest the boundary the Tissot measure also flags as a local maximum. This is three independently

Table 103.1: Three distortion measures for rHEALPix, by region. $\omega$ from Table 31.1; ring-placement ratio from Table 63.1 (deviation from 1 shown); IPQ (refined, $n = 128$) from Chapter 99.

| Region | Tissot $\omega$ | Ring-placement ratio (res. tested) | IPQ (refined) |
|---|---|---|---|
| Equatorial belt (quad), mid-latitude ($\phi \approx 20$–$25°$) | minimum, $\approx 2°$ | — | — |
| Equatorial belt, resolution-0 quad cells | $9.38°$ at $\phi = 0$ | — | $0.94954$ (below flat ceiling) |
| Belt/cap boundary ($\phi \approx 40$–$45°$) | maximum, $\approx$ 21–23° | $+26.6\%$ (res. 2), $-16.0\%$ (res. 3), $-7.5\%$ (res. 4) | — |
| Polar cap, resolution-0 cap cells | 13.8–19.2° | — | $1.20481$ (above flat ceiling) |

computed, three differently defined quantities pointing at the same qualitative region of the grid, which is a stronger form of corroboration than any one of them alone – but it remains qualitative agreement on *where*, not a quantitative agreement on *how much*, and no attempt is made here to turn it into one.

One thing the comparison does *not* show, and this is worth stating plainly rather than leaving implicit: the Tissot $\omega$ minimum at $\phi \approx 20$–$25°$ has no counterpart measurement at that specific latitude in either the ring-placement or IPQ tables, both of which report resolution-0 cell-level or ring-level figures rather than a continuous curve. Whether a genuinely fine-grained discrete measurement at that exact latitude would also show a local optimum, or whether the coarse resolution-0 cells simply average over it, is not established here and would require a resolution sweep of the discrete measures at a matching angular fineness to the continuous one – not attempted in this volume.

## 103.3 Closing the efficiency thread: what query cost does and does not track

Part III opened by asking whether the same distortion-versus-effort trade-off recurs at the level of computation, not just geometry. Across all three grids given from-scratch operational-timing treatment in this volume (H3, S2, rHEALPix), the answer is the same in kind though not in detail: per-operation query cost (hierarchy traversal, neighbour lookup) does *not* track storage cost, which grows geometrically with resolution for all three grids alike. H3's two core operations were flat within measurement noise across six resolutions; S2 reproduced that flat pattern for two of its three operations, with its encoding step flat where H3's showed a small monotonic rise (a genuine structural difference between a fixed-width Hilbert-style index and a per-digit positional one); rHEALPix showed a third, distinct shape on every operation checked, none matching either other grid's pattern exactly. No single query-cost law holds across all three implementations – each library's own internal data structure choices, not the grids' underlying distortion properties, appear to set the shape of its own query-cost curve. This is consistent with this chapter's broader point: the measures this volume has assembled – local (Tissot), finite-scale (ring-placement, IPQ), and operational (query timing) – are related in the sense of describing the same five constructions, but none is shown here to determine another's value, and none should be assumed to generalize past the specific library, machine, and resolution range this volume discloses for it.

**To Complete / Verify**

The efficiency-versus-accuracy question this chapter closes is closed only for the three grids this volume has given operational-timing treatment (H3, S2, rHEALPix); DGGRID and OpenEAGGR remain entirely untimed. The qualitative three-way agreement in Table 103.1 is reported for rHEALPix only. **Partly resolved by Chapter 102:** a resolution-swept, refinement-checked IPQ curve has since been produced for S2 (Table 102.2) and compared against the (by then corrected, Chapter 32) Tissot field (Table 102.3); the two independently agree that S2's worst-behaved region is the immediate neighborhood of the cube corners, located directly by finding the single lowest-IPQ cell at level 4 and confirming its position against the exact cube-corner latitude. **Resolved by Chapter 104:** the ring-placement side has since been run for S2 across the same level-0–6 range used for the IPQ curve above, so all three measures now share a common level range for this grid, as they already did for rHEALPix.

Chapter 104

# The Missing Side, Supplied: S2's Ring-Placement Law Across the Same Levels as Its IPQ Curve

## 104.1 Scope: closing Chapter 103's remaining gap

**Methodological Note**

**What was left open, precisely.** Chapter 103 closed its Tissot/IPQ side for S2 by pointing to Chapter 102, which supplied a refined, resolution-swept IPQ curve at levels 0–6 (native `CellId` enumeration). It left the ring-placement side open, noting that Chapter 63 had already run the ring-placement check for S2 – but only at levels 4–6 ($K = 6, 10, 16$), chosen there to match H3's resolutions 2–4 by cell count, not to match Chapter 102's own level range. This chapter reruns Chapter 63's identical method (`code/s2_ring_placement_law_check.py`'s `run_for_level()`, unchanged) at levels 0–3 as well, so that S2, like rHEALPix already had in that synthesis, finally has all three measures – Tissot, ring-placement, IPQ – available across a common level range rather than three different ones.

**Why levels 0–1 needed a different $K$ than levels 4–6, disclosed rather than smoothed over.** Chapter 62's own rule was to pick $K$, per resolution, as the largest value keeping the polar-most band's real cell count above zero. Applied honestly to S2's coarsest levels, that rule does not yield a neat, level-independent progression: level 0 has only 6 cells globally (one per cube face), so no $K$ above 2 leaves a non-degenerate ring structure at all; level 1 has 24 cells and the same $K = 2$ ceiling; level 2 admits $K = 6$ and level 3 admits $K = 8$ before the polar band empties. $K$ therefore runs $2, 2, 6, 8, 6, 10, 16$ across levels 0–6 – not monotonic, because the underlying grid's discrete cube-corner structure at coarse levels is not monotonic either. This chapter reports the sequence as found rather than picking a smoother-looking but less-honest alternative. Full code: `code/s2_ring_placement_full_sweep.py` (identical `run_for_level()` to `code/s2_ring_placement_law_check.py`, just called across seven configurations instead of three); full output: `data/s2_ring_placement_full_sweep_results.json`. Levels 4–6 reproduce Table 63.1's S2 row exactly ($0.9361$, $0.8306$, $0.9573$) – included here again only so the full seven-level sequence sits in one table, not to claim a new result at those three levels.

Table 104.1: S2 ring-placement law check, levels 0–6, native enumeration. $n_{K-1}$ is the real cell count landing in the polar-most band. Levels 4–6 reproduce Table 63.1 exactly. Source: `code/s2_ring_placement_full_sweep.py`, `data/s2_ring_placement_full_sweep_results.json`.

| level | $K$ | cells | $n_{K-1}$ | $\rho_{K-1}$ (real) | $\tan(\pi/4K)$ | ratio | best-fit $\beta$ |
|---|---|---|---|---|---|---|---|
| 0 | 2 | 6 | 2 | 0.500000 | 0.414214 | 1.2071 | 1.0 |
| 1 | 2 | 24 | 8 | 0.500000 | 0.414214 | 1.2071 | 1.0 |
| 2 | 6 | 96 | 8 | 0.269528 | 0.131652 | 2.0473 | 1.0 |
| 3 | 8 | 384 | 8 | 0.129966 | 0.098491 | 1.3196 | 1.0 |
| 4 | 6 | 1,536 | 48 | 0.123241 | 0.131652 | 0.9361 | 1.0 |
| 5 | 10 | 6,144 | 64 | 0.065368 | 0.078702 | 0.8306 | 1.0 |
| 6 | 16 | 24,576 | 120 | 0.047030 | 0.049127 | 0.9573 | 1.0 |

## 104.2 Results: the full sweep

Two things follow from having all seven levels in one table that could not be said from three.

**Key Result**

The $\beta = 1$ minimum-absolute-difference finding (Sections 62.3.1 and 63.2) is not an artifact of the three resolutions previously tested: it holds at every one of the seven S2 levels checked here, coarse and fine alike, including the two degenerate-$K$ configurations (levels 0–1) where the "ring" structure itself is barely present. Whatever produces the equiangular exponent's specific fit – unproven, candidate-explained only via statistical cancellation (Section 62.4) – is not sensitive to how finely the sphere has been partitioned.

**Key Result**

Absolute agreement with the graticule's $\beta = 1$ prediction is not monotonic in resolution, and is markedly worse at the coarse end than anywhere Table 63.1 previously reported. Levels 0–1 overshoot by 21%; level 2 overshoots by a full 105%, the largest deviation from unity found anywhere in this volume's ring-placement work; level 3 overshoots by 32%; only from level 4 onward does the ratio settle within the 4–17% band Chapter 63 already characterized as S2's typical range. The direction also flips once, from overshoot (levels 0–3) to undershoot (levels 4–5) before drifting back toward unity at level 6 – a sign change this chapter's data shows but does not explain, left as this chapter's own open item below.

**To Complete / Verify**

This chapter closes the level-range gap Chapter 103 left open, but does not explain *why* levels 2–3 show the largest deviations from the $\beta = 1$ prediction, nor why the deviation's sign flips between level 3 and level 4. A candidate mechanism – that S2's cube-face seams contribute proportionally more to the polar-most band's cell population at coarse levels, before the quadtree subdivision dilutes their share – is offered here only as a conjecture, not tested; doing so would require isolating each polar-band cell's originating face and re-running the count face by face, which this chapter did not do.

Chapter 105

# DGGRID: The Missing Fourth Real Trade-off, and a Different Cost Model

## 105.1 Scope: the gap Section 114.5 names twice

Chapter 114 discloses, in its own ledger of what has not yet been attempted, that H3, S2, and rHEALPix have each received a real, from-scratch distortion-versus-storage measurement (Chapters 93, 95 and 97) and a real operational-timing measurement (Chapters 94, 96 and 98), while DGGRID has received only the two-resolution geometric reproduction of Chapter 15 – not the resolution-sweep trade-off or timing treatment given the other three. This chapter closes both items for DGGRID's ISEA7H, using the same from-source build already established in Chapter 15 (v9.03b, commit `04bf5ed`), extended here to a wider resolution sweep and to a batch address-conversion timing test.

**Methodological Note**

**What this chapter does not claim.** DGGRID's cost model turns out, once measured, to be genuinely different in kind from H3's, S2's, and rHEALPix's – not merely offset by a constant. Rather than force DGGRID's numbers into the same per-call microsecond table used for the other three (Tables 94.1, 96.1 and 98.1), which would misrepresent how DGGRID is actually used, this chapter measures DGGRID on its own terms (Section 105.4) and reports the architectural difference itself as a finding.

## 105.2 Method: distortion versus storage

**Methodological Note**

**What was done, precisely, and its limits.** `GENERATE_GRID` was run for whole-earth ISEA7H at resolutions 1–6 (resolution 6, 1,176,492 cells, took 45.0 seconds and produced a 325 MB GeoJSON file – the practical ceiling for wall-clock generation time; resolution 7 was not attempted), emitting every cell's boundary as GeoJSON, exactly as in Chapter 15 but for four more resolutions than that chapter's two. For each cell, area and perimeter were computed with `pyproj`'s `Geod.polygon_ area_perimeter` on the WGS84 ellipsoid – the same engine Chapter 15 already cross-checked there against an independent closed-form spherical-excess formula and found to agree closely, so it is used alone here rather than duplicated. Normalized area (area divided by that resolution's mean cell area) and the isoperimetric quotient $\text{ipq} = 4\pi \cdot \text{area}/\text{perimeter}^2$ are reported per resolution as $\sigma$(norm. area), norm. area min/max, and mean/std/min/max of ipq – the same statistics Chapters 95 and 97 report for S2 and rHEALPix, making Table 105.1 directly comparable to Tables 95.1 and 97.1's columns. Storage cost is the fixed-width SEQNUM index, $\lceil \log_2(\text{num cells}) \rceil$ bits, the

same convention used there. Full code: `code/dggrid_real_tradeoff.py`; full output: `data/dggrid_real_tradeoff_results.json`.

Table 105.1: DGGRID ISEA7H, resolutions 1–6: distortion cost $\sigma$(norm. area) and mean ipq against storage cost (bits, fixed-width SEQNUM index). Source: `code/dggrid_real_tradeoff.py`, `data/dggrid_real_tradeoff_results.json`.

| res | #cells | bits | $\sigma$(norm. area) | na min | na max | ipq mean |
|---|---|---|---|---|---|---|
| 1 | 72 | 7 | 0.10179 | 0.7693 | 1.0541 | 0.91522 |
| 2 | 492 | 9 | 0.02335 | 0.8531 | 1.0148 | 0.89837 |
| 3 | 3,432 | 12 | 0.01745 | 0.7335 | 1.0311 | 0.89792 |
| 4 | 24,012 | 15 | 0.00507 | 0.8602 | 1.0092 | 0.89643 |
| 5 | 168,072 | 18 | 0.00534 | 0.7322 | 1.0308 | 0.89678 |
| 6 | 1,176,492 | 21 | 0.00403 | 0.8618 | 1.0090 | 0.89637 |

## 105.3 What the table shows: a sharp drop, then saturation – and an odd/even wrinkle

$\sigma$(norm. area) falls by a factor of 4.4 from resolution 1 to 2, then keeps falling but far more slowly, ending at 0.00403 by resolution 6 – a factor of 25 over six resolutions rather than a smooth exponential trend across the whole range. This is the same qualitative shape Section 114.3 already reports for H3 (a sharp early drop followed by saturation, against unbounded storage growth) and now a third scheme showing it, alongside S2 (Chapter 95) and rHEALPix (Chapter 97). ipq mean is close to flat across the whole sweep (0.915 at resolution 1, settling to 0.896–0.898 from resolution 2 onward) – ISEA7H's hexagons keep a roughly constant average compactness once past the coarsest, pentagon-dominated resolution, unlike its area uniformity, which keeps improving.

**Key Result**

Normalized-area minimum does not decrease monotonically with resolution: it alternates between two bands, roughly 0.73 (odd resolutions 1, 3, 5) and roughly 0.86 (even resolutions 2, 4, 6). This parity split is consistent with ISEA7H's aperture-7 construction alternating orientation between successive resolutions – the same combinatorial fact already surfaced, in a different guise, in Chapter 25's discussion of aperture-7's two sub-lattice classes – but that connection is observed here, not derived: no closed-form argument is given for why the minimum specifically should track the parity split, only that it does, at all six resolutions measured. Whether it holds beyond resolution 6 is not tested.

## 105.4 Method: batch address-conversion timing

**Methodological Note**

**Why this is not the same protocol as Chapters 94, 96 and 98, and what was done instead.** Those three chapters time single in-process function calls (`h3-py`, `s2sphere`, and `rhealpixdggs` are all Python libraries called directly). DGGRID exposes no equivalent: it is a command-line executable configured by a metafile, and its only point-address-

conversion operation, `TRANSFORM_POINTS`, takes a whole file of points per subprocess invocation, not one point per call. There is no way to time "one DGGRID lookup" without also timing process start-up, metafile parsing, and DGG initialization, so this chapter times what DGGRID actually offers: whole-batch `GEO` → `SEQNUM` conversion, wall-clock timed with `time.perf_counter()` around each full subprocess call. Two sweeps were run: (a) resolutions $1, 3, 5, 7, 9$ at a fixed batch of $100{,}000$ points, and (b) batch sizes $1{,}000$ / $10{,}000$ / $100{,}000$ / $1{,}000{,}000$ at a fixed resolution $(5)$. Points are sampled area-uniformly on the sphere (lat $=$ $\arcsin(U(-1,1))$, lon $= U(-180, 180)$, seed $42$), not uniform in $(\text{lat}, \text{lon})$, which would over-sample the poles. Each configuration was run once: a full subprocess call already costs seconds to minutes, so run-to-run jitter is a much smaller fraction of the total than for a single microsecond -scale in-process call, and repeating dozens of multi-second-to-minute subprocess runs, as Chapter 94's $21 \times 20{,}000$ -call protocol does for microsecond calls, was not affordable here – a real limitation, disclosed rather than hidden, of the single-run figures below. Same machine class as Chapters 94, 96 and 98: single-core container (`nproc`=1), Intel Xeon @ 2.80GHz. Full code: `code/dggrid_timing.py`; full output: `data/dggrid_timing_results.json`.

Table 105.2: DGGRID `TRANSFORM_POINTS`, $100{,}000$ points fixed, resolution varied. Source: `code/dggrid_timing.py`.

| res | elapsed (s) | $\mu$s/point |
|---|---|---|
| 1 | 3.919 | 39.19 |
| 3 | 4.279 | 42.79 |
| 5 | 4.340 | 43.40 |
| 7 | 4.225 | 42.25 |
| 9 | 4.187 | 41.87 |

| #points (res 5) | elapsed (s) | $\mu$s/point |
|---|---|---|
| 1,000 | 0.047 | 46.78 |
| 10,000 | 0.427 | 42.68 |
| 100,000 | 4.278 | 42.78 |
| 1,000,000 | 43.314 | 43.31 |

## 105.5 What the timing shows: flat in resolution, linear in batch size

**Key Result**

DGGRID's per-point address-conversion cost is, within this single-run measurement's noise, independent of resolution ($39$–$43\,\mu$s/point across nine orders of magnitude in cell count, resolutions $1$ through $9$) and independent of batch size once past the smallest batch tested ($42$–$47\,\mu$s/point from $1{,}000$ to $1{,}000{,}000$ points). Fitting elapsed time as $c + n \cdot k$ against the two most different batch sizes gives $k \approx 43.3\,\mu$s/point and a fixed overhead $c \approx 3.5$ ms – small enough that even the $1{,}000$-point batch's per-point figure is within $10\%$ of the $1{,}000{,}000$-point batch's, and negligible next to the multi-second totals at $100{,}000$ points and above.

This is a different shape from all three in-process libraries. H3 (Table 94.1), S2 (Table 96.1), and rHEALPix (Table 98.1) each show resolution-dependent per-call cost for at least one operation (most visibly rHEALPix's `neighbors` call, which grows from $242\,\mu$s at resolution $0$ to a peak of $388\,\mu$s at resolution $3$); DGGRID shows none, for the one operation it exposes. Whether that is because `TRANSFORM_POINTS`'s internal point-location step is genuinely resolution-independent, or because it is dominated by per-point I/O and text parsing that swamps a smaller resolution-dependent term, is not distinguished here – no instrumentation internal to DGGRID's own source was added to separate the two, and doing so would go beyond what this chapter set out to measure.

## 105.6 What this resolves, and what remains

This chapter closes Section 114.5's two DGGRID items together: DGGRID now has a resolution-sweep distortion-versus-storage measurement (Table 105.1, directly comparable in column format to S2's and rHEALPix's) and an operational-timing measurement (Table 105.2, not directly comparable to the other three's per-call figures, for the architectural reasons stated in Section 105.4, but measured on DGGRID's own actual cost model rather than left unmeasured). OpenEAGGR remains the one scheme among Kmoch et al. (2022)'s five with neither a resolution-sweep trade-off nor an operational-timing chapter of its own – Chapter 16's two-resolution snapshot is, like Chapter 15's was before this chapter, a reproduction rather than a sweep. The parity wrinkle in Section 105.3's normalized-area minimum is reported, not explained, and remains open. Whether DGGRID's resolution-independent per-point timing is a property of address conversion specifically, or would also hold for `GENERATE_GRID`'s own per-cell cost, is untested here and is a natural next item.

Chapter 106

# HTM: A Fifth Real Trade-off, and the Cheapest Storage Cost Yet

## 106.1 Scope: the gap this volume's own accounting names

H3, S2, rHEALPix and DGGRID have each received a real, from-scratch distortion-versus-storage measurement across a resolution sweep (Chapters 93, 95, 97 and 105). HTM, reproduced from scratch in Chapter 39, has not: that chapter reports shape statistics (area ratio, area scatter, ipq, angle bounds) at nine depths, which already contains everything a real trade-off table needs except the explicit storage-cost column. This short chapter supplies that column and reads the same table Chapter 39 already computed as a trade-off, exactly as Chapter 105 did for a table it inherited from an earlier chapter rather than recomputing from zero.

**Methodological Note**

**Method.** No new computation is performed in this chapter: the depth-indexed statistics of Table 39.1 are reused unchanged, with storage cost taken as $\lceil \log_2(\#\text{trixels}) \rceil$ bits per trixel address, the same fixed-width-index convention used in Chapters 93, 95, 97 and 105 (this ignores HTM's own variable-length digit-string path names, which this volume's from-scratch reproduction uses internally but which are not directly bit-comparable to a fixed-width index without a further encoding choice this chapter does not make; see Section 106.3). Full code: `code/htm_independent_reproduction.py`; full output: `data/htm_independent_reproduction_results.json`.

## 106.2 The table, read as a trade-off

Table 106.1: HTM distortion cost ($\sigma$(norm. area), mean ipq) against storage cost (bits, fixed-width trixel index), depths 1–9. Same underlying data as Table 39.1. Source: `code/htm_independent_reproduction.py`, `data/htm_independent_reproduction_results.json`.

| depth | #trixels | bits | $\sigma$(norm. area) | ipq mean |
|---|---|---|---|---|
| 1 | 32 | 5 | 0.23316 | 0.6428 |
| 2 | 128 | 7 | 0.24211 | 0.5922 |
| 3 | 512 | 9 | 0.24264 | 0.5800 |
| 4 | 2,048 | 11 | 0.24268 | 0.5770 |
| 5 | 8,192 | 13 | 0.24268 | 0.5763 |
| 6 | 32,768 | 15 | 0.24268 | 0.5761 |
| 7 | 131,072 | 17 | 0.24268 | 0.5761 |
| 8 | 524,288 | 19 | 0.24268 | 0.5760 |
| 9 | 2,097,152 | 21 | 0.24268 | 0.5760 |

**Key Result**

HTM's trade-off curve has a shape this volume has not seen in any of the four schemes already measured: $\sigma$(norm. area) does not fall with added storage at all past depth $2$ — it is already at $0.24211$ by depth $2$ ($7$ bits) and at $0.24268$ by depth $9$ ($21$ bits), a change of under $0.3$ percent across seven further doublings of storage. H3, S2, rHEALPix and DGGRID (Sections 105.3 and 114.3) all show a sharp early drop in $\sigma$(norm. area) followed by slower saturation; HTM shows no drop phase at all, only saturation, because its area irregularity is fixed entirely at the second subdivision level (Section 38.3) and is then reproduced self-similarly, not accumulated or smoothed, at every level after. Extra storage buys HTM finer spatial resolution at essentially unchanged relative area uniformity – a genuinely different trade-off shape, not merely a flatter version of the other four.

## 106.3 A caveat on the storage comparison itself

The fixed-width bit counts in Table 106.1 are directly comparable in magnitude to Tables 93.1, 95.1, 97.1 and 105.1 only if all five schemes are given the same fixed-width-index treatment, which is what this chapter does; but HTM's own native addressing (a root octant letter plus a base-4 digit string, one digit per level) is not fixed-width across depths sharing the same table, since a depth-$d$ path is exactly $d$ digits long regardless of which trixel it names, unlike a SEQNUM- or S2CellId-style fixed-width index. Whether HTM's native variable-length path, treated as a self-delimiting prefix code, would beat or lose to the fixed-width figure at a given depth is not computed here and is left open, alongside the neighbour- finding item already flagged in Chapter 39.

Chapter 107

# Timing HTM's Operations: Descent Cost, and One Open Item

## 107.1 Scope

Chapters 94, 96 and 98 time three in-process operations each for H3, S2 and rHEALPix: point encode, parent lookup, and a $k = 1$ neighbour ring. Chapter 105 timed DGGRID on its own, batch-subprocess terms instead, since a single in-process call is not something DGGRID exposes. HTM's from-scratch reproduction (Chapter 39) is, like H3, S2 and rHEALPix, an in-process Python implementation, so the same per-call protocol applies here; but only two of the three operations can be measured honestly, for reasons stated in Section 107.2.

**Methodological Note**

**Method.** Two operations are timed, at depths $1, 3, 5, 7, 9$, with the same warm-up-then-median-of-21-trials-of-5,000-calls protocol used throughout Part III: (a) `locate`, point $\rightarrow$ trixel path, HTM's analogue of `latlng_to_cell` – implemented as descent from the correct root octant, testing each candidate child via a point-in-spherical-triangle test (three same-side-of-great-circle-plane checks via cross and dot products); and (b) `one_level_children`, generating an already-located trixel's four children by edge-midpoint bisection – the cheapest possible expansion step and a lower bound on the cost any neighbour-adjacent query would have to clear. Machine: same single-core container as Chapters 94, 96 and 98. Full code: `code/htm_operational_timing.py`; full output: `data/htm_operational_timing_results.json`.

## 107.2 The third operation this chapter does not measure

Chapters 94, 96 and 98 each time a $k = 1$ neighbour ring as their third operation. This chapter does not, and says so rather than substituting a proxy for it: Chapter 39's from-scratch implementation covers coordinate-to-trixel descent and top-down subdivision, both purely local operations within a single root-to-leaf path, but not HTM's neighbour-finding algorithm, which every real HTM implementation handles as separate machinery because two trixels can be geometric neighbours while belonging to different root octants and therefore to unrelated digit-string paths – the same kind of cross-tree stitching problem this volume has already met, in a different guise, wherever a grid's addressing scheme and its geometric adjacency structure diverge (Chapter 25). Implementing it was outside this pass's scope; the two operations measured below are a lower bound on HTM's per-call cost, not a complete comparison to the other three schemes' three-operation tables.

## 107.3 Results

Table 107.1: HTM operational timing, depths 1–9: median wall-clock per-call cost ($\mu$s) of point-locate descent and one-level child generation. Source: `code/htm_operational_timing.py`, `data/htm_operational_timing_results.json`.

| depth | locate ($\mu$s) | one_level_children ($\mu$s) |
|---|---|---|
| 1 | 5.448 | 1.507 |
| 3 | 10.164 | 1.432 |
| 5 | 18.182 | 1.457 |
| 7 | 24.085 | 1.461 |
| 9 | 30.187 | 1.463 |

**Key Result**

`locate`'s cost grows roughly linearly with depth ($5.4 \rightarrow 30.2$ $\mu$s across depths 1 to 9, a growth factor of 5.5 across an eightfold increase in depth, consistent with $O(\text{depth})$ descent rather than a constant- or logarithmic-cost lookup), because each additional level requires one more three-sided point-in-triangle test against the previous level's four freshly bisected children – unlike H3's `latlng_to_cell`, S2's `S2CellId.from_lat_lng`, and rHEALPix's coordinate encoders, all of which resolve a coordinate to a deep cell in close to constant time via direct closed-form index arithmetic rather than a per-level geometric test (Chapters 94, 96 and 98). `one_level_children`, by contrast, is flat at 1.4–1.5 $\mu$s regardless of depth, as expected for an operation whose cost (three midpoint computations) does not depend on how deep in the tree it is called.

This is a real architectural difference, not an implementation shortcoming specific to this from-scratch pass: HTM's descent-based design, which makes coordinate encoding a per-level geometric test, is the direct price of the same pole-free, chart-free construction that Chapter 38 credits with HTM's original motivation (avoiding the lat/lon singularity for database range queries). The other three schemes buy constant-time encoding by building index arithmetic directly on top of a chart (a cube face and its own local coordinates, in S2's and rHEALPix's cases); HTM's chart-free construction, the same property that made this chapter's own area formula need to be chart-free too (Section 39.2), is what makes its encode operation depth-dependent instead.

Chapter 108

# OpenEAGGR: A Sixth Real Trade-off, Rebuilt From Source

## 108.1 Scope: extending a two-point sample, as Chapter 105 did for DGGRID

Chapter 16 reproduced OpenEAGGR's ISEA3H geometry from scratch at exactly two resolutions (2 and 4), the same starting point Chapter 15 had left for DGGRID before Chapter 105 extended it to a full resolution sweep. This chapter performs the same extension for OpenEAGGR: resolutions 1–6, using the same enumeration and dual-method area/perimeter measurement (spherical-excess and `pyproj`/WGS84) Chapter 16 already established, with `pyproj`/WGS84 reported as the primary figure here, matching the convention Chapters 93, 95, 97 and 105 already use for H3, S2, rHEALPix and DGGRID.

**Methodological Note**

**Rebuilding OpenEAGGR from source.** The `libeaggr.so` build Chapter 16 used did not persist afterward (only that chapter's *output* data did). Producing this chapter's wider sweep therefore required rebuilding the library first. The riskaware-ltd/open-eaggr source (v2.0 tag) was cloned fresh; its Linux build ships prebuilt static/shared GDAL 20 and PROJ 4.9 libraries vendored directly in the repository (`EAGGR/3rdParty/lib,bin,include`), so no GDAL/PROJ build-from-source was needed, only the library's own 35 C++ files (the same file count Chapter 16 reports). The Eclipse-CDT project ships no standalone Makefile, so the 35 files were compiled directly: `g++ -std=c++14 -fPIC -shared`, with include paths covering the project root, every `Src/API` subdirectory, and the vendored GDAL/OGR/PROJ headers, linked against `libgdal.so.20/libproj.so.9`, plus a system `libboost-dev` install (Boost.Geometry is used internally but not vendored). C++14 was required, not the C++11 first attempted: Boost 1.83 (this container's available version) refuses to compile Boost.Geometry under C++11. The resulting library reports `EAGGR_GetVersion() = "v2.0"`, and, run against Chapter 16's own resolution-2 and resolution-4 figures as a consistency check before any new computation, reproduces both exactly ($\sigma$(norm. area)$=$ 0.0077309 at resolution 2 and 0.0033366 at resolution 4, matching that chapter's values to the digit) – treated here as confirmation that this is a faithful rebuild, not a different library version producing coincidentally similar numbers. Full code: `code/eaggr_real_tradeoff.py`; full output: `data/eaggr_real_tradeoff_results.json`.

Table 108.1: OpenEAGGR ISEA3H, resolutions 1–6: distortion cost $\sigma$(norm. area) and mean ipq (`pyproj`/WGS84 method) against storage cost (bits, fixed-width cell index). "skipped" counts cells where `EAGGR_ConvertDggsCellOutlineToShapeString` returned a degenerate or NaN-vertex outline, disclosed and excluded, as in Chapter 16, not interpolated. Source: `code/eaggr_real_tradeoff.py`, `data/eaggr_real_tradeoff_results.json`.

| res | #cells | bits | $\sigma$(norm. area) | ipq mean | skipped |
|---|---|---|---|---|---|
| 1 | 140 | 8 | 0.00694 | 0.94184 | 0 |
| 2 | 560 | 10 | 0.00773 | 0.87686 | 0 |
| 3 | 2,220 | 12 | 0.00308 | 0.85509 | 0 |
| 4 | 7,500 | 13 | 0.00334 | 0.83183 | 80 |
| 5 | 24,600 | 15 | 0.00265 | 0.82046 | 260 |
| 6 | 77,300 | 17 | 0.00267 | 0.81111 | 1,320 |

## 108.2 Results

**Key Result**

Two things stand out against the four schemes already measured (Sections 105.3 and 114.3). First, ISEA3H's $\sigma$(norm. area) never shows the sharp early drop H3, S2, rHEALPix and DGGRID all share: it starts already low ($0.00694$ at resolution $1$) and stays within a narrow $0.0026$–$0.0077$ band across the whole six-resolution sweep – area uniformity essentially bought at resolution $1$ already, not earned gradually. Second, the same odd/even-style pairing Section 105.3 reports for DGGRID appears here too, but with period $2$ rather than DGGRID's period $1$: resolutions $\{1, 2\}$ pair near $0.007$, $\{3, 4\}$ pair near $0.0032$, $\{5, 6\}$ pair near $0.0027$, each pair-boundary showing a level drop rather than each individual resolution. This is consistent with ISEA3H's aperture-3 hexagonal construction repeating orientation every two resolutions rather than every one (Chapter 25), exactly the kind of aperture-dependent periodicity Section 105.3 already flagged as observed, not derived, for DGGRID's aperture-7 case; the same caveat applies here.

ipq mean falls steadily and monotonically from $0.942$ at resolution $1$ to $0.811$ at resolution $6$ – unlike DGGRID's roughly flat ipq mean past its coarsest resolution (Section 105.3), ISEA3H's hexagons keep losing compactness as resolution increases, the opposite direction from what its improving-then-plateauing area uniformity might suggest. The skipped-cell count also grows with resolution ($0$ at resolutions $1$–$3$, $1{,}320$ of $77{,}300$, or $1.7$ percent, at resolution $6$): the NaN-vertex artifact Chapter 16 first disclosed at resolutions $2/4$ becomes more frequent, not less, at finer resolutions, since finer cells sit closer, in absolute count, to the icosahedron's twelve singular vertices for a fixed base-cell layout.

Chapter 109

# Timing OpenEAGGR's Operations, and a Crash Caught Along the Way

## 109.1 Scope

Chapters 94, 96, 98 and 107 time in-process operations for H3, S2, rHEALPix and HTM. OpenEAGGR's rebuilt library (Chapter 108) is likewise in-process (a `ctypes` wrapper over `libeaggr.so`), so the same per-call protocol applies, timing OpenEAGGR's own three closest analogues to the encode/parent/neighbour-ring triple used for H3, S2 and rHEALPix: `EAGGR_ConvertPointsToDggsCells` (point encode, resolution controlled indirectly via an `accuracy_m2` parameter, calibrated per target resolution by bisection search – see Section 109.2), `EAGGR_GetDggsCellParents` (parent), and `EAGGR_GetDggsCellSiblings` (OpenEAGGR's own same-resolution neighbour query, the direct analogue of `grid_disk(k=1)`).

**Methodological Note**

**Method.** Same warm-up-then-median-of-21-trials-of-2,000-calls protocol used throughout Part III, on the same single-core container. For each target resolution in Table 108.1, `accuracy_m2` is bisection-searched (on a $\log_{10}$ scale, 60 iterations) until `point_to_cell` returns a cell at that exact resolution; the calibrated value is then used for the timed encode calls, and the cell it returns is used for the timed parent/siblings calls. Full code: `code/eaggr_operational_timing.py`; full output: `data/eaggr_operational_timing_results.json`.

## 109.2 A crash caught, not hidden

The first run of this chapter's script timed `siblings` in the same process as everything else, exactly like the other operations. At resolution 5 it segfaulted the entire Python process outright – not a raised exception, since a C library corrupting its own out-parameter count crashes past anything `ctypes` or Python can catch. Tracing the cause (Section 109.3) showed this was not an isolated fluke of that one resolution: OpenEAGGR's `siblings` call returns exactly 6 neighbours at resolutions 1–2 (the geometrically correct answer for a hexagonal cell with no special boundary condition), 14 at resolution 3, and exactly 15 – this wrapper's `MAX_SIBLING_CELLS` buffer capacity, copied from OpenEAGGR's own header constant – at resolutions 4 and 6, strongly suggesting the true sibling count exceeds the buffer at those resolutions and is silently truncated to it, while at resolution 5 the same situation instead overruns the buffer's bounds outright and crashes. This was addressed, for this chapter's purpose, by running the `siblings` timing for each resolution in its own isolated subprocess, so that a crash at one resolution does not discard the timing already collected for the other five; resolution 5's `siblings` figure is therefore reported as missing, not estimated or interpolated, in Table 109.1. This is a genuine defect (or, at minimum, an undocumented resolution-dependent behaviour change)

in OpenEAGGR v2.0 itself, caught by this volume's own practice of timing every operation directly rather than trusting a library's documented complexity, in the same spirit as the `EAGGR_DeallocateString` segfault Chapter 16 already disclosed for a different call in the same library.

## 109.3 Results

Table 109.1: OpenEAGGR operational timing, resolutions 1–6: median wall-clock per-call cost ($\mu$s) of point encode, parent lookup, and same-resolution siblings. Resolution 5's siblings figure is missing because the call crashes the process at that resolution (Section 109.2), not because it was not attempted. Source: `code/eaggr_operational_timing.py`, `data/eaggr_operational_timing_results.json`.

| res | encode ($\mu$s) | parents ($\mu$s) | siblings ($\mu$s) | #siblings returned |
|---|---|---|---|---|
| 1 | 3.432 | 3.299 | 19.218 | 6 |
| 2 | 3.610 | 3.295 | 18.494 | 6 |
| 3 | 3.589 | 5.029 | 91.542 | 14 |
| 4 | 5.945 | 5.222 | 89.348 | 15 |
| 5 | 3.519 | 5.186 | *crashed* | — |
| 6 | 3.516 | 5.176 | 87.489 | 15 |

**Key Result**

Encode and parent costs are both flat with resolution, at 3.3–5.9 $\mu$s and 3.3–5.2 $\mu$s respectively across the whole sweep – close to constant-time, unlike HTM's $O(\text{depth})$ `locate` (Chapter 107), and in the same low-single-digit-microsecond range this volume has already measured for H3's `latlng_to_cell` and `cell_to_parent` (Chapter 94). Siblings cost jumps sharply, from $\approx$ 19 $\mu$s at resolutions 1–2 (returning 6 neighbours) to $\approx$ 88–92 $\mu$s at resolutions 3, 4 and 6 (returning 14 or 15) – a roughly $4.5\times$–$4.8\times$ cost increase tracking almost exactly the $2.3$–$2.5\times$ increase in the number of cells actually returned, i.e. per-neighbour cost itself is also higher at finer resolutions, not merely the count. Whether this reflects a genuinely more expensive per-neighbour computation at finer resolutions, or is a further symptom of the same buffer/count irregularity that crashes resolution 5, is not established here and is left as a further open item alongside the crash itself.

Chapter 110

# GBT's Trade-off: When Distortion Cost Is Exactly Zero

## 110.1 Scope: the trade-off collapses to one dimension

Every scheme examined so far in Part III — H3, S2, rHEALPix, DGGRID, HTM, OpenEAGGR (Chapters 93, 95, 97, 105, 106 and 108) — is a *spherical* DGGS, and each of those chapters reports a two-dimensional trade-off: storage cost against distortion cost, the latter falling as resolution increases but never reaching zero because no cell forced onto a curved surface can be a Euclidean regular hexagon exactly. GBT is different, and Chapter 46 already established why: it is an unbounded planar tessellation, never projected onto a sphere at all, and every cell it produces — checked there, not assumed — is an exact Euclidean regular hexagon with ipq $= \sqrt{3}\pi/6 = 0.906900\ldots$ and $\sigma(\text{norm. area}) = 0$, at *every* aggregation level. This chapter does not recompute that figure; doing so would duplicate Chapter 46 under a new name. What it does instead is examine what GBT's trade-off looks like once the distortion dimension is held fixed at its floor: a pure storage-cost profile, plus one quantitative cross-check against a spherical scheme that shares GBT's own digit assignment.

**Methodological Note**

**What was done.** For digit-string lengths $n = 1, \ldots, 6$, every one of the $7^n$ candidate GBT addresses is enumerated and converted to Cartesian coordinates via Chapter 46's own verified forward map, to check — rather than assume — that all $7^n$ addresses land on distinct points (no collisions from the specific digit geometry). Storage cost is then reported two ways: the information-theoretic minimum $\lceil n \log_2 7 \rceil$ bits, and the practical fixed-width packing this volume's other tables use for their own schemes (here, 3 bits per digit, since a GBT digit's 7 values fit in 3 bits with one code unused). Finally, GBT's exact algebraic cell-count growth of $7^n$ is checked against H3's own previously published, independently measured cell counts at resolutions 0–5 (Table 93.1), since H3 uses the same aperture-7 digit assignment (Chapter 25) but on an icosahedron with twelve pentagonal singularities. Full code: `code/gbt_real_tradeoff.py`; full output: `data/gbt_real_tradeoff_results.json`.

## 110.2 Results

All $7^n$ candidate addresses were confirmed distinct at every $n$ from 1 to 6 (zero collisions in every case), so the growth is exact, not approximate. The two bit-cost columns agree exactly through $n = 5$ and diverge by exactly one bit at $n = 6$ (17 against 18): $\lceil 6 \log_2 7 \rceil = 17$, but 6 digits at 3 bits each is 18, the first point in this range where fixed-width per-digit packing stops being information-theoretically tight.

Table 110.1: GBT storage cost by digit-string length $n$: exact address count $7^n$ (verified collision-free against the geometric forward map), information-theoretic minimum bits, and practical 3-bits/digit packing. Distortion columns are not new computations: they are Chapter 46's established exact constants, repeated here for the trade-off comparison. Source: `code/gbt_real_tradeoff.py`.

| $n$ digits | #cells ($7^n$) | bits (info. min.) | bits (3/digit) | $\sigma$(norm. area) | ipq mean |
|---|---|---|---|---|---|
| 1 | 7 | 3 | 3 | 0 | 0.90690 |
| 2 | 49 | 6 | 6 | 0 | 0.90690 |
| 3 | 343 | 9 | 9 | 0 | 0.90690 |
| 4 | 2,401 | 12 | 12 | 0 | 0.90690 |
| 5 | 16,807 | 15 | 15 | 0 | 0.90690 |
| 6 | 117,649 | 17 | 18 | 0 | 0.90690 |

**Key Result**

GBT is the one scheme in this volume's whole cost-of-distortion comparison where the trade-off is not really a trade-off: distortion cost sits at its exact floor from $n = 1$ onward and never moves, so the entire storage-versus-distortion frontier that Chapters 93, 95, 97, 105, 106 and 108 each trace as a curve collapses, for GBT, to a single horizontal line: storage grows geometrically while shape quality is already perfect and stays perfect. This is not a gap in the analysis to be filled with a more careful measurement — it is the correct description of an addressing algebra that was never asked to cover a sphere in the first place, and Chapter 47's own finding that the identical exact ceiling holds for MGBT's aperture-3 planar hexagons too (its Section 47.4) confirms this is a property of central-place hexagon tessellation on the plane generally, not an accident of GBT's particular aperture.

Table 110.2: GBT's exact $7^n$ growth against H3's own independently measured cell-count growth factor per resolution step (Table 93.1), since H3 uses the same aperture-7 digit assignment on an icosahedron. Source: `code/gbt_real_tradeoff.py`.

| Resolution step | H3 measured factor | Gap from exact 7 | Relative gap |
|---|---|---|---|
| $0 \to 1$ | 6.90164 | 0.09836 | 1.405% |
| $1 \to 2$ | 6.98575 | 0.01425 | 0.2036% |
| $2 \to 3$ | 6.99796 | 0.00204 | 0.0291% |
| $3 \to 4$ | 6.99971 | 0.00029 | 0.0042% |
| $4 \to 5$ | 6.99996 | 0.00004 | 0.00059% |

**Key Result**

H3's measured growth factor approaches GBT's exact 7 from below at every step, and the relative gap shrinks by almost exactly one order of magnitude per resolution step ($1.4\% \to 0.20\% \to 0.029\% \to 0.0042\% \to 0.00059\%$). This is the direct, quantitative signature of H3's twelve pentagonal base cells (Chapter 25): they are a fixed count sitting at a fixed twelve icosahedral vertices, so their contribution to the total cell count — and hence their drag on the growth factor away from GBT's clean planar 7 — shrinks geometrically as the denominator $7^n$ grows, while their absolute number never changes. GBT, having no vertices to be singular at, supplies the exact reference rate that H3's own empirical figures can be measured against; without GBT's closed form, this convergence rate would have to be inferred from the H3 numbers alone rather than checked against a known exact target.

**To Complete / Verify**

This chapter's storage comparison uses digit-string length $n$ as GBT's resolution proxy, but H3's own resolution numbering starts from $122$ base cells at resolution $0$, not $7^0 = 1$; the growth-factor comparison above is therefore about the *ratio* between consecutive resolutions, not about matching absolute cell counts between the two schemes at a given resolution label, and Table 110.2 should be read that way. A cleaner joint framework relating GBT digit count directly to H3, DGGRID and OpenEAGGR resolution labels — all three of which also embed the same aperture-7 or mixed-aperture digit assignment inside a base-cell subdivision this chapter does not model — is left open.

Chapter 111

# Timing GBT's Own Operations, and a Second Bug Caught Along the Way

## 111.1 Scope: what GBT has to time, and what it does not

Chapters 94, 96, 98, 107 and 109 time an encode/parent/neighbour-ring triple for H3, S2, rHEALPix, HTM and OpenEAGGR. GBT cannot supply that same triple: Chapter 46 disclosed openly that the reverse, Cartesian-to-GBT conversion was never reproduced, because van Roessel's own indexing matrix $K$ is only partially specified in the source text available to that chapter — so there is no `point_to_cell`-equivalent to time here, and this is stated rather than filled with a proxy, in the same spirit as Chapter 107's disclosed gap for HTM's own neighbour-finding.

What GBT's own verified code does support is timed instead: the forward digit-to-Cartesian *decode* map, digit-by-digit *addition with carry propagation* — GBT's own signature operation, with no analogue timed for any other scheme in this volume — and the six single-digit additions that stand in for a *neighbour ring*, at digit-string lengths $n = 1, 2, 4, 6, 8$.

**Methodological Note**

**A second bug caught, this time in this volume's own code.** Chapter 46's `gbt_add_with_carry` was verified there against exactly one worked example, van Roessel's own $44 + 56 = 563$. Running that same function on randomly generated pairs of addresses six digits or longer, as this chapter's timing protocol does, raised a `KeyError` on a nontrivial fraction of trials. The cause: at a position with an incoming carry, the function computes a carry-out from adding that carry into the digit, and a second carry-out from adding the other addend's digit, then combines the two carry-outs with plain Python integer addition (`carry += digs[1]`) rather than GBT's own addition table. This is only valid if both carry-out digits happen to be $0$ or $1$ – true of the single two-digit worked example Chapter 46 checked, but not true in general: two GBT digits summed can themselves carry, and doing so with ordinary integer addition instead of the table produces an invalid "digit" (values up to 7 were observed) that no subsequent table lookup can resolve. This chapter implements a corrected adder, `gbt_add_with_carry_fixed`, which folds any number of pending carries into each position sequentially via the addition table itself rather than by integer arithmetic, and verifies it two ways before using it for any timing: against van Roessel's own worked example (reproduced exactly, $563$), and independently against Chapter 46's geometric forward map, summing $2{,}000$ random address pairs at digit-lengths 1–4 both via the fixed adder and via direct coordinate addition, with zero mismatches. Chapter 46's own claim is not wrong — its one worked example genuinely does reproduce $563$ — only narrower in scope than a reader might assume; this is disclosed as an erratum to that chapter rather than a correction quietly folded in unannounced. Full code: `code/gbt_operational_timing.py`; full output: `data/gbt_operational_timing_results.json`.

## 111.2 Results

Table 111.1: GBT operational timing, digit-string lengths 1–8: median wall-clock per-call cost ($\mu$s) of forward decode, addition with carry, and six-neighbour generation from a random address versus a constructed worst-case address (all digits chosen, by direct inspection of the addition table, to carry when incremented). "Carried positions" is the number of positions that actually receive a nonzero incoming carry when 1 is added to the worst-case address, traced directly rather than inferred from output length. Source: `code/gbt_operational_timing.py`, `data/gbt_operational_timing_results.json`.

| $n$ digits | decode ($\mu$s) | add ($\mu$s) | nbrs, random ($\mu$s) | nbrs, worst-case ($\mu$s) | carried positions |
|---|---|---|---|---|---|
| 1 | 0.457 | 2.400 | 22.11 | 21.94 | 1 |
| 2 | 0.633 | 6.755 | 26.37 | 33.18 | 2 |
| 4 | 1.098 | 10.444 | 41.09 | 53.98 | 4 |
| 6 | 1.257 | 13.871 | 56.72 | 75.63 | 6 |
| 8 | 1.877 | 12.314 | 73.42 | 96.66 | 8 |

**Key Result**

Decode is, and stays, the cheapest operation measured here by roughly an order of magnitude (0.46–1.9 $\mu$s across the whole range), growing slowly and roughly linearly with digit count, as expected for a single pass summing $n$ precomputed offset vectors. Addition grows faster with $n$ (2.4 $\rightarrow$ 13.9 $\mu$s from $n$=1 to $n$=6, a 5.8$\times$ increase for a 6$\times$ increase in digit count) since every position potentially requires resolving a pending carry via a further table lookup; the slight drop at $n$=8 (12.3 against $n$=6's 13.9) falls within this protocol's trial-to-trial noise elsewhere in this volume (Chapter 109) and is not treated as a reversal of the trend.

The worst-case-address neighbour timing is consistently and increasingly more expensive than the random-address figure, by 0% at $n$=1 (both cases carry only into a possible one further digit) widening to 32% at $n$=8 (96.7 against 73.4 $\mu$s) — and the "carried positions" column confirms why directly, rather than leaving it as a plausible-looking correlation: a worst-case address, built from digits the addition table itself marks as carry-producing under +1, propagates a carry through *every* one of its $n$ positions, exactly matching $n$ in each row of Table 111.1. A random address, by contrast, mixes carrying and non-carrying digits and so triggers a much shorter chain on average.

**Key Result**

GBT's own addition is the one operation in this volume's whole operational-timing survey with a genuine, demonstrated $O(n)$ worst case in the digit-string length — not merely a theoretical possibility noted in passing, but a cost gap between a constructed worst-case address and a random one that widens systematically as $n$ grows, traced back to an exact count of carrying positions. This contrasts with H3's, S2's, DGGRID's and OpenEAGGR's roughly flat, close-to-constant-time point operations (Chapters 94, 96, 105 and 109), and is a direct structural consequence of GBT being a genuine positional number system with carry propagation — the same property that made this chapter's own bug possible in the first place, and that neither a fixed-index spherical DGGS nor a single-pass coordinate descent (HTM, Chapter 107) has any equivalent of.

**To Complete / Verify**

Two items were left open. **(i)** This chapter times addition and its worst case using addresses constructed for that purpose; it does not attempt to characterize the *distribution* of carry-chain lengths over uniformly random address pairs (only the flat mean already visible in the random-address column). Chapter 112 closes this with a $20{,}000$-sample study per digit length, finding a mean chain length approaching roughly $47\%$ of $n$ and a full-cascade probability that falls from over a third at $n{=}1$ to under $2\%$ at $n{=}8$. **(ii)** The corrected `gbt_add_with_carry_fixed` introduced here is not back-ported into Chapter 46's own code file in this pass; both versions currently coexist in `code/`, with this chapter's disclaimer as the pointer between them, rather than silently overwriting a previous chapter's verified artifact.

Chapter 112

# The Shape of a Carry Chain: Sampling GBT's Addition Cost

## 112.1 Scope: what Chapter 111 left as a flat mean

Chapter 111's todobox left one item open: that chapter times GBT addition on a random address and, separately, on a constructed worst-case address that forces a full-length carry cascade, but never characterizes the *distribution* of carry-chain lengths a uniformly random pair of addresses actually produces — only the flat per-call mean already visible in its random-address timing column. This chapter runs that sampling study.

**Methodological Note**

No new adder is introduced. This chapter reuses, unmodified, Chapter 111's own corrected `gbt_add_with_carry_fixed` and its carry-tracing variant `gbt_add_with_carry_traced` (which returns the number of positions actually receiving a nonzero incoming carry, not merely whether the result is one digit longer than the inputs), together with the addition table derived in Chapter 46. Before sampling, the imported table is checked against a fresh derivation to guard against a stale import. Full code: `code/gbt_carry_chain_distribution.py`; full output: `data/gbt_carry_chain_distribution_results.json`.

## 112.2 Method

For each digit-string length $n \in \{1, 2, 4, 6, 8\}$ — Chapter 111's own set — $20{,}000$ pairs of independent, uniformly random $n$-digit addresses are generated and added with the traced adder, recording the exact number of carried positions for each pair. This is a distinct random seed from Chapter 111's own timing runs, not a replay of the same samples.

## 112.3 Results

The single-position case $(n = 1)$ can be checked exactly rather than only by sampling: a carry occurs on a single digit-pair addition exactly when the addition table's entry for that pair is two digits long, which direct enumeration of all $49$ ordered pairs in the addition table shows is true for $18$ of them, giving an exact carry probability of $18/49 = 0.36735$ — matching this section's Monte Carlo estimate $(0.363$ over $20{,}000$ trials$)$ to within sampling noise, and cross-checking the sampler itself independently of the timing protocol it is built from.

Table 112.1: Distribution of carry-chain length (number of positions receiving a nonzero incoming carry) over 20,000 random address pairs per digit-string length $n$. "Full cascade" is the fraction of pairs whose carry chain reaches all $n$ positions, matching Chapter 111's constructed worst case exactly; "zero carry" is the fraction with no carry propagation at all. Source: `code/gbt_carry_chain_distribution.py`, `data/gbt_carry_chain_distribution_results.json`.

| $n$ | mean | median | $p_{90}$ | $p_{99}$ | max | full cascade | zero carry |
|---|---|---|---|---|---|---|---|
| 1 | 0.363 | 0 | 1 | 1 | 1 | 36.3% | 63.7% |
| 2 | 0.825 | 1 | 2 | 2 | 2 | 22.3% | 39.8% |
| 4 | 1.797 | 2 | 3 | 4 | 4 | 8.9% | 16.1% |
| 6 | 2.786 | 3 | 5 | 6 | 6 | 3.3% | 6.4% |
| 8 | 3.731 | 4 | 6 | 8 | 8 | 1.5% | 2.8% |

**Key Result**

Mean carry-chain length grows with $n$, but sublinearly relative to the worst case: as a fraction of $n$ itself, the mean rises from 36.3% ($n$=1) through 41.2%, 44.9%, 46.4% to 46.6% ($n$=8), approaching roughly 47% of the digit-string length rather than continuing to grow toward 100%. *Full*-cascade pairs — the event Chapter 111's worst-case address constructs deliberately — become rapidly rarer as $n$ grows, from over a third of random pairs at $n$=1 (where "full cascade" merely means the one position carries) to under 2% at $n$=8. This is the quantitative shape behind Chapter 111's own qualitative observation that random-address timings stayed well below the worst-case figures at every $n$: not because carrying is rare in absolute terms, but because a chain reaching *every* position specifically is rare, and increasingly so as the address gets longer.

**Key Result**

This closes the gap left open in Chapter 111: GBT addition's cost genuinely does have a demonstrated $O(n)$ worst case (Table 111.1, worst-case column), but the *typical* case, quantified here for the first time in this volume, grows at under half that rate on average and puts successively less of its probability mass near the worst case as $n$ increases — the two figures this chapter and Chapter 111 give together, worst-case and typical-case, are both needed to characterize GBT's addition cost; neither alone would be a complete picture, and Chapter 111 disclosed this gap explicitly rather than letting the flat random-address timing column stand in for a full characterization.

**To Complete / Verify**

This chapter samples uniformly random digit addresses, matching Chapter 111's own random-address protocol. It does not characterize carry-chain length under a non-uniform, geographically-derived distribution of addresses (i.e. the actual distribution GBT cells would have if sampled from real coastline or population data, as several DGGS reproduction chapters elsewhere in this volume do for cell counts) – GBT's disclosed lack of a Cartesian-to-GBT conversion (Chapter 46) means no such geographically-grounded sample of GBT addresses can be constructed in this volume, and this is left open for the same structural reason.

Chapter 113

# An Operational Resolution Crosswalk Across Five DGGS, and Why GBT Stays Out of It

## 113.1 Scope: what "unifying resolution numbering" can and cannot mean here

Earlier chapters of Part III treated H3, S2, rHEALPix, DGGRID, and OpenEAGGR one at a time, each with its own resolution index running over its own range with its own aperture. Nothing in that treatment lets a reader answer a simple cross-scheme question directly: which S2 level is comparable, in physical cell size, to H3 resolution 7? This chapter builds that correspondence for the five schemes that have an actual global cell inventory, and is explicit about a stronger claim it does *not* make: it does not unify the schemes' underlying index *arithmetic* (their different digit alphabets, different apertures –7, 3, 4, 9 – and different base polyhedra), only their physical scale. Section 113.4 explains why this volume's own GBT work cannot be given even that weaker treatment.

**Methodological Note**

**Method.** For each scheme, the number of cells $N(r)$ at resolution $r$ was obtained either from a real library call (`h3.get_num_cells` for H3; `rhealpixdggs`'s own `num_cells` for rHEALPix, $N_side = 3$) or a closed form already implicit in the scheme's own published construction (S2: $6 \cdot 4^{\ell}$ for level $\ell$, six root faces under a quaternary quadtree), or from this volume's own already-executed pipeline outputs (`data/dggrid_real_tradeoff_results.json` for DGGRID/ISEA7H, resolutions 1–6; `data/eaggr_isea3h_growth_results.json`'s deduplicated counts for OpenEAGGR/ISEA3H, resolutions 0–9). No new library or binary was invoked for DGGRID or OpenEAGGR in this chapter – the sandbox limitation already disclosed elsewhere in this volume ( neither `dggrid` nor `libeaggr.so` is installed here) was worked around by re-using results this volume had already produced and cited, not by re-deriving them. The effective mean cell size is defined as $D(r) = \sqrt{A_{\oplus}/N(r)}$, with $A_{\oplus} = 4\pi R_{\oplus}^2$ and $R_{\oplus} = 6371.0088\,\text{km}$ (IUGG mean radius); this is an area-normalization proxy, not a claim that any of these schemes' cells are literally that size or that shape at every point (Chapters 31 and 36 of this volume have already shown, for three of the five, that they are not, uniformly). Full code: `code/unified_resolution_framework.py`; full output: `data/unified_resolution_framework_results.json`.

## 113.2 The aperture check: does $D(r)$ actually behave geometrically?

Before trusting $D(r)$ as an interpolation variable, its own internal consistency was checked: for an exact aperture-$k$ scheme, successive sizes should satisfy $D(r)/D(r+1) = \sqrt{k}$ at every $r$, since $N(r+1) = k \cdot N(r)$ exactly.

Table 113.1: Empirical $D(r)/D(r{+}1)$ ratios against the scheme's nominal $\sqrt{\text{aperture}}$. Source: `code/unified_resolution_framework.py`.

| Scheme | nominal $\sqrt{\text{aperture}}$ | observed range across resolutions |
|---|---|---|
| H3 (aperture 7) | 2.6458 | 2.627 (res. 0→1) settling to 2.64575 by res. 3 |
| DGGRID/ISEA7H (aperture 7) | 2.6458 | 2.614 (res. 1→2) settling to 2.6457 by res. 4 |
| S2 (aperture 4) | 2.0000 | exactly 2.0000 at every level (closed-form quadtree) |
| rHEALPix (aperture 9) | 3.0000 | 3.162 (res. 0→1) settling to 3.00000 by res. 4 |
| OpenEAGGR/ISEA3H (aperture 3) | 1.7321 | $2.646 \to 2.0 \to 1.99 \to \cdots \to 1.743$, **not converging to** 1.732 within res. 0–9 |

**Key Result**

Four of the five schemes converge to their nominal $\sqrt{\text{aperture}}$ quickly (by resolution 3–4) and some exactly by construction (S2, trivially, since $6 \cdot 4^{\ell}$ is exactly geometric at every step, not just asymptotically). OpenEAGGR's ISEA3H is the exception: its deduplicated cell counts (already present in this volume's own `data/eaggr_isea3h_growth_results.json`, not recomputed here) grow by a ratio that starts near 7 (resolution $0 \to 1$: $20 \to 140$ cells), passes through 4 (resolution $1 \to 2$: $140 \to 560$), and is still at only $D$-ratio 1.743 (i.e. not yet at the nominal 1.732) by resolution 8–9 – consistent with an aperture-3 hexagonal icosahedral scheme whose early resolutions are dominated by pentagon correction terms that only asymptotically wash out, not with an error in either that chapter's or this chapter's arithmetic. **This chapter's crosswalk table therefore treats ISEA3H's low resolutions (below about 5) as less reliable interpolation anchors than the other four schemes', and reports this explicitly rather than smoothing over it.**

## 113.3 The crosswalk table

Using each scheme's own two extreme available resolutions to fix the log-linear relationship $D(r) = D(r_0) \cdot k^{-(r-r_0)/2}$ (exact for S2; an extrapolation, flagged above, for ISEA3H at the coarse end), the resolution index nearest a series of round target cell sizes was computed by inverting that relationship, i.e. $r = r_0 - 2\log(D_{\text{target}}/D(r_0))/\log k$:

**Key Result**

Read as a practical instrument rather than a theorem: at the $\sim 10$ km scale often used for regional analysis (the same order of magnitude as Chapters 79 and 91's own case studies), H3 resolution 5, DGGRID/ISEA7H resolution 7, S2 level 10, and rHEALPix resolution 6 are its closest same-scale counterparts – a correspondence not previously stated anywhere in this volume, where each scheme's resolution had been discussed only in its own chapter's

Table 113.2: Resolution index (real-valued; round to the nearest integer for an actual library call) whose effective cell size is closest to each target. Source: `data/unified_resolution_framework_results.json`.

| target size | H3 | DGGRID/ISEA7H | S2 | rHEALPix | OpenEAGGR/ISEA3H† |
|---|---|---|---|---|---|
| 1000 km | 0.7 | 2.0 | 3.2 | 2.0 | 2.9 |
| 300 km | 2.0 | 3.2 | 4.9 | 3.1 | 5.1 |
| 100 km | 3.1 | 4.4 | 6.5 | 4.1 | 7.1 |
| 10 km | 5.5 | 6.7 | 9.8 | 6.2 | 11.3 |
| 1 km | 7.8 | 9.1 | 13.2 | 8.3 | 15.5 |

†Extrapolated from resolutions 0 and 9 using the observed, not-yet-converged ratio; per Section 113.2, read this column as indicative, not to the same precision as the other four.

own terms.

## 113.4 Why GBT is not in the table

van Roessel (1988)'s planar aperture-7 digit assignment, independently reproduced in Chapter 45, was verified there to share H3's *local* step structure exactly: both are aperture-7 hexagonal constructions with the same $19.10661°$ dual-hexagon rotation already computed in `data/gbt_geometry_check_results.json`. This chapter adds one further, independent confirmation of that number, not previously stated in closed form in this volume: the rotation angle satisfies

$$\theta_{\text{ap7}} = \arcsin\sqrt{\tfrac{3}{28}} = 19.106605350869096°,$$

which matches the already-computed value to every digit reported (an exact match to the 15 significant figures both computations carry, not an approximation). Chapters 12 and 45's own aperture-7 reproduction of H3 and this volume's fresh derivation of $\theta_{\text{ap7}}$ from the exact value $3/28$ therefore triangulate on the same constant from three independent directions.

That local correspondence, however, is where the unification stops. Sahr (2011)'s icosahedral MGBT extension of GBT – which would wrap the planar aperture-7 table onto the sphere, giving it a bounded global cell count $N(r)$ comparable to H3's – is explicitly not attempted in Chapters 45 and 47, both of which state plainly that they reproduce the digit-assignment table without the icosahedral tiling, for lack of a located primary source sufficient to re-derive Sahr's specific tiling choice independently. Without that tiling, GBT as this volume has actually built it has no global $N(r)$, and therefore no $D(r)$: it is a planar system that could in principle be extended over an unbounded area, not (yet, in this volume) a bounded global grid. Including it in Table 113.2 would require inventing a global cell count that does not exist in anything computed here – exactly the kind of step this volume's no-invented-data discipline rules out. GBT's row in the unified picture is therefore: *locally identical to H3, globally undefined*, which is itself the honest answer to the question this chapter opened with, for that one scheme.

**To Complete / Verify**

The crosswalk of Table 113.2 unifies scale, not index arithmetic: it does not attempt a common addressing scheme, a shared digit alphabet, or a formal isomorphism across the four apertures (7, 7, 4, 9) actually tabulated, only a numerical correspondence in effective cell size. OpenEAGGR/ISEA3H's column is flagged as the least reliable, for the reason given in Section 113.2: no further resolutions were computed in this pass to firm it up (doing so

would require the DGGRID/OpenEAGGR binaries this sandbox does not have installed, per the scope note in the method box above). Closing GBT's gap fully would require locating a primary source for Sahr's MGBT icosahedral tiling and reproducing it independently, which Chapters 45 and 47 already declined to do without one and which this chapter does not revisit.

Chapter 114

# Where This Volume Stands: An Honest Accounting

## 114.1 What this chapter is, and is not

Chapter 37 of Monograph 4 closed that volume with an explicit diagnosis of what had been proved, what had failed by extrapolation, and what remained open, reporting negative results with the same care as positive ones. This chapter does the same for the present, still-growing volume, at an earlier stage of its construction. It is not a synthesis of a finished argument; it is an honest ledger of a partial one.

## 114.2 What stands on a fully independent derivation

Three results in this volume are exact, self-contained mathematical propositions, verified numerically to machine precision, and not contingent on any external empirical claim:

- Proposition 1.2 (Chapter 1): the exact cell-area law $\rho_i = \cos\big((i+\frac{1}{2})\Delta\phi\big)/\cos(\Delta\phi/2)$ for the $K$-ring discrete graticule, with the classical cos $\phi$ didactic argument recovered exactly as its $K \to \infty$ limit.
- Propositions 58.1 and 58.3 (Chapter 58): the exact boundary law $\rho_{\min}(K) = \tan(\pi/4K) = \Theta(K^{-1})$ and the exact uniform bulk error bound $\Theta(K^{-2})$, together showing the pole is a genuine boundary singularity in the same structural sense used throughout *No Map Is Best*.
- Proposition 78.2 (Chapter 78): the monotonicity of a fixed-width band's internal area-ratio spread in its central latitude, giving a model-independent, structural reason to expect regional optimization to pay off more at high latitude.
- Proposition 59.3 and Theorem 59.6 (Chapter 59): the dual, equal-area scheme's own exact boundary law (angular width diverging relative to the mean at rate $\Theta(K^{1/2})$), and a general impossibility theorem proving that no ring-partition scheme, of any kind, can be simultaneously equiangular and equal-area for $K \geq 2$ — the non-toy foundation underneath Chapter 92's Pareto illustration.
- Propositions 60.3 and 60.5 (Chapter 60): the exact boundary-ratio formula for a one-parameter family of general (non-equiangular) ring placements, exactly recovering Proposition 58.1 at $\beta = 1$, and a proof that the boundary rate undergoes a phase transition at the closed-form threshold $\beta = 1/2$ — vanishing, converging to a finite constant, or diverging, depending on which side of the threshold the placement falls.

A fourth item is not a closed-form proposition but a fully executed, independently reproduced empirical measurement rather than a toy illustration: Chapter 93's finding that H3's

normalized-area distortion, computed from scratch across resolutions 0–5, drops sharply over the first two resolution levels and then saturates to four significant figures, while its operational storage cost keeps growing geometrically without bound. Unlike the three propositions above, this is not a mathematical theorem; it is a directly measured fact about one real DGGS implementation, reusing Chapter 12's already-verified code path rather than any new or unverified method. A fifth, similarly empirical, item: Chapter 94's timed measurement that per-operation query cost (hierarchy traversal, neighbour lookup) stays flat across the same resolution range where storage cost explodes, run twice independently to separate a real pattern from single-run measurement noise. A sixth item, of a different character again: Chapter 14's systematic test of two candidate mechanisms behind the long-standing S2 resolution-to-level ambiguity, one parameter-free and externally corroborated (date-line-crossing removal), one explicitly disclosed as reverse-fit rather than derived (a polar-cap threshold) — together narrowing, though not completely closing, an ambiguity left open since Chapter 12. A seventh item extends the fourth: Chapter 95 repeats Chapter 93's full distortion-versus-storage measurement independently on S2 (levels 0–8, native `s2sphere` enumeration, the library's own area and vertex-angle perimeter geometry), finding a qualitatively different low-end shape from H3 – exact area-uniformity at levels 0–1 followed by a *rise* to a saturation plateau, rather than H3's steady fall – while confirming, now from an independent computation rather than Kmoch & Vasilyev (2022)'s transcribed figures alone, that S2 saturates later and at a higher $\sigma$ than H3 does; a small, unexplained offset of $0.0018$–$0.0026$ against those transcribed figures is disclosed there, and Chapter 95 itself later rules out one of its two named candidate mechanisms (date-line-crossing inclusion accounts for under 2% of it) without identifying the true cause. An eighth item extends the fifth: Chapter 96 times S2's own encode, hierarchy-step, and neighbour-query operations across the same level range, run twice independently; two of the three reproduce H3's flat pattern (at different absolute constants, and with S2's encoding operation flat where H3's showed a small monotonic rise, a genuine structural difference between the two indexing schemes), while the third (`get_all_neighbors`) shows a falling-then-plateauing shape confirmed in both runs but not mechanistically explained.

These three results share a proof style deliberately continuous with Monographs 1–4: closed-form derivation first, numerical verification second, with the numerical check never used to *discover* the result, only to confirm it.

## 114.3 What rests on cited external sources, and how firmly

Four external sources anchor the empirical and applied claims of this volume. Their editorial records have now all been independently verified against primary bibliographic metadata (publisher pages, institutional publication lists, or the issue's own table of contents), resolving what were open items in the working draft:

- Kmoch et al. (2022) (Section 1.3): fully verified editorial record. The underlying normalized-area and compactness *values*, flagged here as not yet reproduced in an earlier draft of this chapter, have since been transcribed verbatim from the authors' own released data supplement (Chapter 11) and, for three of the five schemes (H3, DGGRID, and OpenEAGGR), independently recomputed from scratch on freshly regenerated cell geometry (Chapters 12, 15 and 16). S2 and rHEALPix have received independent, from-scratch treatment of their own real-tradeoff and operational-timing behaviour (Chapters 95 and 96 and Chapters 97 to 101 respectively), though not this specific two-resolution area/compactness comparison. All five schemes Kmoch et al. (2022) studied have therefore now received some form of from-scratch, independent treatment in this

volume.

- Author (s) (Chapter 71): editorial record verified except for the authors' names, which have not been located in any publicly accessible excerpt consulted so far; the RMSE-reduction range and band-wise variance claim are taken directly from the publisher's abstract.
- Kerkovits (2026) (Chapter 75): fully verified editorial record, including issue placement; the sphere's derivation itself is not yet reproduced, and the reported distortion figures are still abstract-level.
- Wecker et al. (2024) (Chapter 92): fully verified editorial record, including full author affiliation.

## 114.4 What is toy, and clearly labelled as such

Chapter 92's Pareto relation (Proposition 92.2) is an internally consistent, exactly provable result about a stylized model built from this volume's own definitions; it illustrates the qualitative shape of Wecker et al. (2024)'s stated tension and nothing more. No numeric constant from that toy model ($\pi^2/32$, or the specific exponents involved) should migrate into a claim about any real DGGS implementation without an explicit, independent re-derivation on real cell geometry. That re-derivation has now been attempted, for H3 alone, in Chapter 93: the real measurement does not simply confirm the toy model's smooth trade-off shape, it complicates it (a sharp early drop in distortion followed by saturation, against unbounded storage growth), which is itself evidence that the toy model's constants must not be read as more than illustrative.

## 114.5 What has failed, or not yet been attempted, by extrapolation

Following Ch. 37 of Monograph 4's discipline of disclosing negative or incomplete results with the same visibility as positive ones:

- Chapter 71 explicitly declines to bridge Proposition 58.1's exact geometric boundary rate with the empirical RMSE-reduction figures of Author (s): the two quantities are not evidently commensurable, and no functional relationship between them is asserted.
- Chapter 78 explicitly declines to claim that Kerkovits (2026)'s specific construction outperforms any specific global DGGS at any specific latitude: Proposition 78.2 establishes only a necessary structural precondition (the available gain grows with latitude), not a sufficient one. A first, deliberately partial step toward the direct comparison has since been taken (Chapter 80, extended to H3 by Chapter 81): S2, rHEALPix, and H3's real within-band area uniformity are now all reported side by side for Kerkovits's own $\Phi_0 = 45°$ band, without being summed or ranked against the rectifying sphere's own linear-scale figure, since the two kinds of quantity are explicitly disclosed there as non-commensurable. H3 shows the largest within-band area ratio of the three ($2.33$–$2.37\times$, against S2's $1.59$–$1.76\times$ and rHEALPix's mechanically trivial $1.000000$), an ordering only partly predicted by the three grids' already-established global area-uniformity ranking (Section 1.3). DGGRID and OpenEAGGR remain absent from this comparison, since both require a from-scratch local build not yet carried out in this volume; a shape/angular metric within the band, and the rectifying sphere's own comparison against authalic

and conformal low-distortion spheres, also remain unattempted. The diagnosis of where regional optimization fails by extrapolation beyond its band, in the spirit of Ch. 37, remains undone.

- Section 58.5's non-equiangular generalization has now been carried out for one explicit family (Chapter 60), and its application to a genuine DGGS cell geometry has now received a first, honestly qualified attempt (Chapter 62): real H3 cell-area mass, binned by centroid latitude into the graticule's own rings, matches Proposition 58.1's $\beta = 1$ prediction to within 3.2% at three resolutions, with $\beta = 1$ the specific best fit among eight scanned exponents rather than an arbitrary one. This item has since been extended, not left at H3 alone: Chapter 63 reproduces the same $\beta = 1$-specific best fit on S2 and rHEALPix as well, at all nine (grid, configuration) points tested in total, while finding that H3's close *absolute* agreement with the idealized law does not generalize (S2 and rHEALPix deviate substantially more, and with inconsistent sign). Chapter 64 has since tested, and ruled out, one specific candidate explanation for that gap (within-band cell-area heterogeneity, correlation $r = -0.619$, wrong sign), leaving the true mechanism – and the sign inconsistency – open, but now with one plausible account eliminated rather than untested, and a shape-based (IPQ) alternative proposed as the next thing to check. That alternative has since been carried out: Chapter 65 finds polar-band mean IPQ correlates with $|\text{ratio} - 1|$ at $r = -0.686$, explaining item (iv)'s cross-grid ordering (H3's more isoperimetric cells track its closer agreement) but not item (v), since mean IPQ is nearly flat across resolution within any single grid while the deviation is not monotonic there. Chapter 67 tested the band's IPQ *variance* instead of its mean, finding a stronger cross-grid correlation ($r = 0.815$) but also a structural reason it cannot explain item (v) either: variance is non-negative by construction and falls monotonically with resolution in all three grids, while the signed deviation does not move monotonically in any of them. Chapter 68 then tested the one candidate not structurally excluded by that argument – IPQ *skewness*, which unlike variance is not sign-bounded – and found it fails empirically rather than structurally: its sign stays fixed within each grid across all three tested resolutions (negative throughout for H3 and S2, positive throughout for rHEALPix), matching the deviation's own sign pattern in only one of the three grids (S2, where the deviation itself never flips), for five sign-matches out of nine points overall, indistinguishable from chance at this $n$. A third, logically independent candidate – that the deviation reflects the graticule boundary's own placement relative to each grid's fixed lattice, rather than any property of the cells – was tested in Chapter 66: shifting the band boundary by 10%–40% of its own angular width changes the ratio by a full order of magnitude, uniformly across all three grids and moderately correlated with baseline deviation ($r = 0.636$), but the effect is smooth and monotonic in the shift, not the sign-flipping jaggedness a genuine per-cell aliasing signature would produce, pointing instead to a geometric property shared by all three grids (the last band's intrinsic angular thinness) rather than a grid-specific lattice interaction. After five independent mechanisms tested across five chapters – area heterogeneity, IPQ mean, IPQ variance, IPQ skewness, and boundary placement – item (v)'s resolution-to-resolution sign flip remains unexplained by any of them; this is reported here as a standing negative result in its own right, not as an item still merely "open."

- Section 60.5's minimality-transition claim (that ring $K - 1$ stops being the global-minimum-area ring exactly at $\beta = 1/2$) is checked only over a finite grid of $K$, not proved for all $K$; this is disclosed explicitly in that section's disclaimerbox rather than presented as a closed theorem.

- Chapter 93's real distortion-versus-storage measurement has now been repeated inde-

pendently for S2 as well (Chapter 95), confirming a qualitatively different pattern from H3, as Chapter 13's transcribed-data comparison had already suggested. rHEALPix has since received its own extensive real-tradeoff and operational-timing measurement (Chapters 97 to 101), and DGGRID has received an independent, from-scratch area/compactness reproduction of its own (Chapter 15), though the latter is the two-resolution comparison format of Chapter 11 rather than the resolution-sweep distortion-versus-storage or operational-timing measurements done for H3, S2, and rHEALPix, so it is not directly comparable to those three chapters' figures. OpenEAGGR has since received the same two-resolution geometric reproduction treatment (Chapter 16), built from its own source (35 C++ files compiled and linked into a from-scratch `libeaggr.so`) rather than a precomputed binary, closing Section 12.3's reproduction item for all five schemes; but, like DGGRID's chapter, it is the two-resolution snapshot format, not the resolution-sweep distortion-versus-storage or operational-timing measurements done for H3, S2, and rHEALPix. The query-cost timing gap has likewise now been closed for S2 (Chapter 96) and rHEALPix (Chapter 98), but not for DGGRID or OpenEAGGR. The small, systematic offset between Chapter 95's own figures and Kmoch & Vasilyev (2022)'s transcribed ones is narrowed (one of two candidate mechanisms is now ruled out) but not fully diagnosed there; an analogous but numerically different unexplained gap (in cell count, and in the resolution-4 normalized-area minimum) is disclosed for DGGRID in Chapter 15 without being diagnosed either.

- Chapter 94's query-cost timings are single-core, single-library, single-machine measurements, explicitly disclosed as not portable in absolute value to any other environment; whether the flat-versus-growing split it finds between query cost and storage cost generalizes to other DGGS, to larger neighbourhoods, or to concurrent workloads is left open there rather than claimed here.

- A separate thread, entirely absent from earlier drafts of this closing chapter, concerns HTM's indicatrix. Chapter 40 found that real HTM trixels neither converge to a single-gnomonic-chart comparison object near a root octant's centre nor grow in area away from it, as that naive comparison predicts – they shrink, to roughly half the centre-trixel area by $25°$–$35°$ out, measured along two bearings on one octant (N0). Chapter 42 extended this to all eight root octants and both azimuthally distinct bearing classes, finding the same shrinkage pattern reproduces identically everywhere, ruling out octant choice as the explanation but leaving the fine azimuthal structure between the two tested bearings unmeasured. Chapter 43 closed that gap with a fine sweep of the fundamental domain and found the curve is *not* smooth between them: area-ratio jumps of up to $55\%$ occur between adjacent $3°$ steps at $z \geq 20°$, consistently across all eight octants, a real consequence of the recursive chord-midpoint construction crossing different trixel boundaries at different azimuths rather than measurement noise (the near-centre case, at $z = 10°$, is smooth to within $1.1\%$). Chapter 44 then closed the first item of that chapter's own todobox by re-sweeping the single largest jump it found ($12°$–$15°$ at octant N1, $z = 20°$) at $0.01°$ resolution – three hundred times finer – and located the crossing precisely at $13.74°$–$13.75°$, confirming it is a single, real, exactly locatable recursive-descent boundary crossing at the very first subdivision level (level $0$ of $11$), not an artefact of the coarser sweep's resolution. Of the $300$ adjacent sub-degree pairs checked across that same $301$-point fine sweep, $36$ show a descent-path divergence at some level, but only the one already identified diverges at level $0$; the rest cluster at the deepest levels tested, consistent with most nearby bearings crossing only fine, late-stage trixel boundaries with a comparatively small effect on area. None of these four chapters attempts a closed-form replacement for the naive comparison object that has been shown to fail; each explicitly declines to curve-fit a several-parameter form to a handful of measured points without

an independent reason to prefer one curve over another, so a positive characterization of HTM's true area-scaling law remains undone, and that chapter's own second todobox item – extending the fine sweep's coverage across $z$ rather than resolution – remains untested.

## 114.6 Honest status of the volume as a whole

As of this revision, the volume comprises 112 chapters across three parts, totalling 418 pages – well past the roughly 150-page, thirty-or-more-chapter scale of Monographs 1–4 (Monograph 4 alone contains 37 chapters). This accounting currently extends to the ring-placement thread through Chapters 65 to 68 and to the HTM-octant thread through Chapters 40 and 42 to 44, both narrated in full above. It does not constitute a full re-audit of the volume: the chapters of Parts II and III (the Kerkovits-band comparisons, the GBT/PYX-IS/EAGGR reproduction and operational-timing chapters, the cost-distortion genealogy and polyhedron-orientation chapters, and others) have not been individually re-checked against the claims made here, and the chapter/page counts above should be read as a snapshot taken at the time of this revision, not a quantity re-verified on every future addition. A growing volume with a closing chapter that narrates only part of it is itself a disclosed risk, not a hidden one: the honest status of any thread not explicitly named above is that this chapter has not re-examined it, and its true current state should be checked directly in that thread's own chapters and their own todobox environments rather than assumed here.

This chapter does not declare the cycle closed. It declares, honestly, where a growing volume currently stands, for the threads this revision was able to bring up to date.

# References

This monograph draws on the classical and information-theoretic apparatus of Monographs 1–4 (collected as *No Map Is Best*), cited here as a whole with chapter numbers given in the text at each point of reuse, and on four externally published sources whose editorial records have each been independently verified against primary bibliographic metadata, as documented at each point of citation and summarized in Chapter 114.

# Bibliography


Kmoch, A., Vasilyev, I., Virro, H., & Uuemaa, E. (2022). *Area and shape distortions in open-source discrete global grid systems*. Big Earth Data, 6(3), 256–275. https://doi.org/10.1080/20964471.2022.2094926

Author(s) to be confirmed (2026). *Hexagonal discrete global grid system enhances model latitudinal spatial generalization and deep learning capabilities*. Big Earth Data. https://doi.org/10.1080/20964471.2026.2666953

Kmoch, A., & Vasilyev, I. (2022a). *Code supplement for open-source DGGS cell geometry comparisons* [Software]. Zenodo. https://doi.org/10.5281/zenodo.5905935; mirrored at https://github.com/LandscapeGeoinformatics/dggs_comparisons.

Kmoch, A., & Vasilyev, I. (2022). *Data supplement for open-source DGGS cell geometry comparisons* [Dataset]. Zenodo. https://doi.org/10.5281/zenodo.6634479

Kerkovits, K. (2026). *Low-Distortion Rectifying Sphere for Regional Maps*. KN – Journal of Cartography and Geographic Information, 76(2). https://doi.org/10.1007/s42489-026-00216-5

Gauß, C. F. (1843). *Erste Abhandlung*. In: Untersuchungen über Gegenstände der Höheren Geodäsie. Wilhelm Engelmann, Leipzig, pp. 3–35.

Kerkovits, K. (2025). *Development of a Low-Distortion Authalic Sphere for the Oblique Azimuthal Equal-Area Map Projection of the Spheroid*. Journal of Spatial Information Science, 2025(30), 117–129. https://doi.org/10.5311/JOSIS.2025.30.402

Sahr, K. (2026). *DGGRID: A discrete global grid generation software* (Version 9.03b) [Software]. https://github.com/sahrk/DGGRID

Wecker, L., Hall, J., & Samavati, F. F. (2024). *Constructing Efficient Mesh-Based Global Grid Systems with Reduced Distortions*. ISPRS International Journal of Geo-Information, 13(11), 373. https://doi.org/10.3390/ijgi13110373

RiskAware Ltd. (2017). *OpenEAGGR: Equal Area Global Grid* (Version 2.0) [Software]. https://github.com/riskaware-ltd/open-eaggr

Snyder, J. P. (1992). *An Equal-Area Map Projection for Polyhedral Globes*. Cartographica: The International Journal for Geographic Information and Geovisualization, 29(1), 10–21. https://doi.org/10.3138/27H7-8K88-4882-1752

Gibb, R. G., Raichev, A., & Speth, M. (2016). *The rHEALPix Discrete Global Grid System*. IOP Conference Series: Earth and Environmental Science, 34, 012012. https://doi.org/10.1088/1755-1315/34/1/012012

Górski, K. M., Hivon, E., Banday, A. J., Wandelt, B. D., Hansen, F. K., Reinecke, M., & Bartelmann, M. (2005). *HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere*. The Astrophysical Journal, 622(2), 759–771. https://doi.org/10.1086/427976

Calabretta, M. R., & Roukema, B. F. (2007). *Mapping on the HEALPix Grid*. Monthly Notices of the Royal Astronomical Society, 381(2), 865–872. https://doi.org/10.1111/j.1365-2966.2007.12297.x

Google (2024). *S2Geometry: S2 Cells* [Software documentation]. https://s2geometry.io/devguide/s2cell_hierarchy.html

Barnes, R. (2019). *Optimal Orientations of Discrete Global Grids and the Poles of Inaccessibility*. International Journal of Digital Earth, 13(7), 803–816 (published online 14 Feb 2019). https://doi.org/10.1080/17538947.2019.1576786

Zhou, J., Ben, J., Wang, R., Zheng, M., Yao, X., & Du, L. (2020). *A Novel Method of Determining the Optimal Polyhedral Orientation for Discrete Global Grid Systems Applicable to Regional-Scale Areas of Interest*. International Journal of Digital Earth, 13(12), 1553–1569 (published online 14 Apr 2020). https://doi.org/10.1080/17538947.2020.1748127

Barnes, R. (2021, comment dated 5 Sep. 2021). Comment on Issue #510, "Orientations," *uber/h3* repository, GitHub. https://github.com/uber/h3/issues/510

White, D., Kimerling, A. J., & Overton, W. S. (1992). *Cartographic and Geometric Components of a Global Sampling Design for Environmental Monitoring*. Cartography and Geographic Information Systems, 19(1), 5–22. https://doi.org/10.1559/152304092783786636

Openshaw, S. (1984). *The Modifiable Areal Unit Problem*. Concepts and Techniques in Modern Geography, No. 38. Geo Books, Norwich.

Openshaw, S., & Taylor, P. J. (1979). *A million or so correlation coefficients: three experiments on the modifiable areal unit problem*. In N. Wrigley (Ed.), *Statistical Applications in the Spatial Sciences* (pp. 127–144). Pion, London.

Gehlke, C. E., & Biehl, K. (1934). *Certain Effects of Grouping Upon the Size of the Correlation Coefficient in Census Tract Material*. Journal of the American Statistical Association, 29(185A), 169–170. https://doi.org/10.1080/01621459.1934.10506247

Robinson, W. S. (1950). *Ecological Correlations and the Behavior of Individuals*. American Sociological Review, 15(3), 351–357.

Fotheringham, A. S., & Wong, D. W. S. (1991). *The Modifiable Areal Unit Problem in Multivariate Statistical Analysis*. Environment and Planning A, 23(7), 1025–1044. https://doi.org/10.1068/a231025

Raposo, P., Robinson, A. C., & Brown, R. (2019). *A Virtual Globe Using a Discrete Global Grid System to Illustrate the Modifiable Areal Unit Problem*. Cartographica: The International Journal for Geographic Information and Geovisualization, 54(1), 51–62. https://doi.org/10.3138/cart.54.1.2018-0015

Nelson, J. K., & Brewer, C. A. (2017). *Evaluating data stability in aggregation structures across spatial scales: revisiting the modifiable areal unit problem*. Cartography and Geographic Information Science, 44(1), 35–50. https://doi.org/10.1080/15230406.2015.1093431

Wagner, R. (2022). *Modifiable Areal Unit Problem — The Spatial Data Scientist's Nightmare* [Blog post]. Medium.

Sahr, K. (2008). *Location Coding on Icosahedral Aperture 3 Hexagon Discrete Global Grids*. Computers, Environment and Urban Systems, 32(3), 174–187. https://doi.org/10.1016/j.compenvurbsys.2007.11.005

Sahr, K. (2011). *Hexagonal Discrete Global Grid Systems for Geospatial Computing*. Archives of Photogrammetry, Cartography and Remote Sensing, 22, 363–376.

Sahr, K. *Icosahedral Modified Generalized Balanced Ternary and Aperture 3 Hexagon Tree*. U.S. Patent 7,876,967 B2, filed Jan. 21, 2005, granted Jan. 25, 2011. https://patents.google.com/patent/US7876967B2/en

Kmoch, A., Sahr, K., Chan, W. T., & Uuemaa, E. (2025). *IGEO7: A New Hierarchically Indexed Hexagonal Equal-Area Discrete Global Grid System*. AGILE GIScience Series, 6, 32. https://doi.org/10.5194/agile-giss-6-32-2025

Christaller, W. (1966). *Central Places in Southern Germany* (C. W. Baskin, Trans.). Prentice-Hall, Englewood Cliffs, NJ. (Original work published 1933, *Die zentralen Orte in Süddeutschland*.)

Bondaruk, B., Roberts, S. A., & Robertson, C. (2020). *Assessing the State of the Art in Discrete Global Grid Systems: OGC Criteria and Present Functionality*. Geomatica, 74(1), 9–30. https://doi.org/10.1139/geomat-2019-0015

Snyder, J. P. (1987). *Map Projections: A Working Manual*. U.S. Geological Survey Professional Paper 1395. U.S. Government Printing Office, Washington, DC.

European Petroleum Survey Group (EPSG); Petrotechnical Open Software Corporation (POSC) (compilers). *Guidance Note Number 7: Coordinate Conversions and Transformations including Formulas*. Mirrored (undated re-publication of a revision whose internal history table ends at "August 2002") at https://gdal.org/proj_list/guid7.html.

Wikipedia contributors (2026). *Transverse Mercator projection* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Transverse_Mercator_projection

Wikipedia contributors (2026). *Robinson projection*; *Winkel tripel projection*; *Winkel projection* [Encyclopedia articles]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Robinson_projection

Karney, C. F. F. (2023). *On Auxiliary Latitudes*. Survey Review. Preprint: https://arxiv.org/abs/2212.05818

Kimerling, A. J., Sahr, K., White, D., & Song, L. (1999). *Comparing Geometrical Properties of Global Grids*. Cartography and Geographic Information Science, 26(4), 271–288. https://doi.org/10.1559/152304099782294186

Carr, D. B., Olsen, A. R., & White, D. (1992). *Hexagon Mosaic Maps for Display of Univariate and Bivariate Geographical Data*. Cartography and Geographic Information Systems, 19(4), 228–236. https://doi.org/10.1559/152304092783721231

Carr, D. B. (1990). *Looking at Large Data Sets Using Binned Data Plots*. Technical Report, Pacific Northwest Laboratory, Richland, WA, for the U.S. Department of Energy, Office of Energy Research. https://doi.org/10.2172/6930282

U.S. Environmental Protection Agency, Environmental Monitoring and Assessment Program (EMAP). *EMAP Master Glossary* [Archived webpage]. https://archive.epa.gov/emap/archive-emap/web/html/mglossary.html

Kawakami, Y., Yuniar, S., & Ma, K.-L. (2024). *HexTiles and Semantic Icons for MAUP-Aware Multivariate Geospatial Visualizations*. arXiv:2407.16897. https://doi.org/10.48550/arXiv.2407.16897

Carr, D. B., Littlefield, R. J., Nicholson, W. L., & Littlefield, J. S. (1987). *Scatterplot Matrix Techniques for Large N*. Journal of the American Statistical Association, 82(398), 424–436. https://doi.org/10.1080/01621459.1987.10478445

Carr, D., Lewin-Koh, N., Maechler, M., Sarkar, D., & Pebesma, E. (2024). *hexbin: Hexagonal Binning Routines* [R package, version 1.28.5]. Comprehensive R Archive Network (CRAN). https://CRAN.R-project.org/package=hexbin

Pędzich, P. (2017). *Image of the World on polyhedral maps and globes*. Polish Cartographical Review, 48(4), 197–210. https://doi.org/10.1515/pcr-2016-0014

Cahill, B. J. S. (1909). *An Account of a New Land Map of the World*. The Scottish Geographical Magazine, 25(9), 449–469. https://doi.org/10.1080/00369220908733981

Cahill, B. J. S. (1932). *The 'Butterfly' Map Projection* [Correspondence]. Nature, 130, 973–974. https://doi.org/10.1038/130973a0

Wikipedia contributors (2026). *Cahill–Keyes projection*; *Waterman butterfly projection*; *Bernard J. S. Cahill* [Encyclopedia articles]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Cahill%E2%80%93Keyes_projection

Tissot, N.-A. (1881). *Mémoire sur la représentation des surfaces et les projections des cartes géographiques*. Paris: Gauthier-Villars.

Papadopoulos, A. (2017). *Nicolas-Auguste Tissot: A Link Between Cartography and Quasiconformal Theory*. Archive for History of Exact Sciences, 71(4), 319–336. https://doi.org/10.1007/s00407-016-0186-z

Miyachi, H., Ohshika, K., & Papadopoulos, A. (2025). *On the Marinus–Ptolemy and Delisle–Euler Conical Maps*. arXiv:2503.22694 [math.HO]. Forthcoming in Gaṇita Bhāratī (Bulletin of the Indian Society for History of Mathematics). https://arxiv.org/abs/2503.22694

Milnor, J. (1969). *A Problem in Cartography*. American Mathematical Monthly, 76(10), 1101–1112.

Dutton, G. (1989). *Planetary Modelling via Hierarchical Tessellation*. Proceedings of the Auto-Carto 9 Conference, Baltimore, MD, 462–471. https://cartogis.org/docs/proceedings/archive/auto-carto-9/pdf/planetary-modeling-via-hierarchical-tessellation.pdf

Dutton, G. (2016). *The Making of a Global Grid — Remembering My Escape from Flatland*. IOP Conference Series: Earth and Environmental Science, 34, 012007. https://doi.org/10.1088/1755-1315/34/1/012007

Raposo, P. (2019). *QTM: A Global Tessellation Generator* [Software repository]. GitHub. https://github.com/paulojraposo/QTM

Kunszt, P. Z., Szalay, A. S., & Thakar, A. R. (2001). *The Hierarchical Triangular Mesh*. In A. J. Banday, S. Zaroubi, & M. Bartelmann (Eds.), *Mining the Sky*, ESO Astrophysics Symposia, 631–637. Berlin, Heidelberg: Springer. https://doi.org/10.1007/10849171_83

Szalay, A. S., Gray, J., Fekete, G., Kunszt, P. Z., Kukol, P., & Thakar, A. R. (2007). *Indexing the Sphere with the Hierarchical Triangular Mesh*. arXiv:cs/0701164 [cs.DB]. Microsoft Research Technical Report MSR-TR-2005-123. https://arxiv.org/abs/cs/0701164

Gibson, L., & Lucas, D. (1982). *Spatial Data Processing Using Generalized Balanced Ternary*. Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing, 566–571.

van Roessel, J. W. (1988). *Conversion of Cartesian Coordinates from and to Generalized Balanced Ternary Addresses*. Photogrammetric Engineering and Remote Sensing, 54(11), 1565–1570. https://www.asprs.org/wp-content/uploads/pers/1988journal/nov/1988_nov_1565-1570.pdf

Mandelbrot, B. B. (1982). *The Fractal Geometry of Nature*, Chapter 6, "Snowflakes and Other Koch Curves," pp. 46–47. San Francisco: W. H. Freeman and Company.

Lucas, D., & Gibson, L. (1970). *A System for Hierarchical Addressing in Euclidean Space*. Unpublished document, pp. 1–11.

White, D. (2000). *Global Grids from Recursive Diamond Subdivisions of the Surface of an Octahedron or Icosahedron*. Environmental Monitoring and Assessment, 64(1), 93–103. https://doi.org/10.1023/A:1006407023786

Bell, S. B. M., Diaz, B. M., Holroyd, F., & Jackson, M. J. (1983). *Spatially Referenced Methods of Processing Raster and Vector Data*. Image and Vision Computing, 1(4), 211–220.

Huang, H.-C., & Cressie, N. (1997). *Multiscale Spatial Modeling*. Proceedings of the Section on Statistics and the Environment, American Statistical Association, Alexandria, VA, pp. 49–54.

Peterson, P. (2011). *Close-Packed, Uniformly Adjacent, Multiresolutional, Overlapping Spatial Data Ordering*. U.S. Patent 8,018,458 B2, PYXIS Innovation Inc. (filed 2004, granted 2011; priority 2003). https://patents.google.com/patent/US8018458B2/en

Fuller, R. B. (1946). *Cartography*. U.S. Patent 2,393,676, filed Feb. 25, 1944, granted Jan. 29, 1946 (expired 1963). https://patents.google.com/patent/US2393676

Vince, A., & Zheng, X. (2009). *Arithmetic and Fourier Transform for the PYXIS Multi-Resolution Digital Earth Model*. International Journal of Digital Earth, 2(1), 59–79. https://doi.org/10.1080/17538940802657694

Kitto, W. Z., Vince, A., & Wilson, D. C. (1994). *An Isomorphism Between the p-Adic Integers and a Ring Associated with a Tiling of N-Space by Permutohedra*. Discrete Applied Mathematics, 52, 39–51. https://people.clas.ufl.edu/avince/files/P-adic.pdf

Féjes Tóth, L. (1964). *Regular Figures*. Pergamon Press, New York.

Sadourny, R., Arakawa, A., & Mintz, Y. (1968). *Integration of the Nondivergent Barotropic Vorticity Equation with an Icosahedral-Hexagonal Grid for the Sphere*. Monthly Weather Review, 96(6), 351–356. https://doi.org/10.1175/1520-0493(1968)096%3C0351:IOTNBV%3E2.0.CO;2

Williamson, D. L. (1968). *Integration of the Barotropic Vorticity Equation on a Spherical Geodesic Grid*. Tellus, 20(4), 642–653. https://doi.org/10.1111/j.2153-3490.1968.tb00406.x

Staniforth, A., & Thuburn, J. (2012). *Horizontal Grids for Global Weather and Climate Prediction Models: A Review*. Quarterly Journal of the Royal Meteorological Society, 138(662), 1–26. https://doi.org/10.1002/qj.958

Williamson, D. L. (1969). *A Comparison of Spherical Grids for Numerical Integration of Atmospheric Models*. Ph.D. thesis, Department of Meteorology, Massachusetts Institute of Technology. https://dspace.mit.edu/handle/1721.1/13773

Williamson, D. L. (1969). *Numerical Integration of Fluid Flow over Triangular Grids*. Monthly Weather Review, 97(12), 885–895. https://doi.org/10.1175/1520-0493(1969)097%3C0885:NIOFFO%3E2.3.CO;2

Sadourny, R., & Morel, P. (1969). *A Finite-Difference Approximation of the Primitive Equations for a Hexagonal Grid on a Plane*. Monthly Weather Review, 97(6), 439–445. https://doi.org/10.1175/1520-0493(1969)097%3C0439:AFAOTP%3E2.3.CO;2

Sadourny, R. (1972). *Conservative Finite-Difference Approximations of the Primitive Equations on Quasi-Uniform Spherical Grids*. Monthly Weather Review, 100(2), 136–144. https://doi.org/10.1175/1520-0493(1972)100%3C0136:CFAOTP%3E2.3.CO;2

Williamson, D. L., Drake, J. B., Hack, J. J., Jakob, R., & Swarztrauber, P. N. (1992). *A Standard Test Set for Numerical Approximations to the Shallow Water Equations in Spherical Geometry*. Journal of Computational Physics, 102(1), 211–224. https://doi.org/10.1016/S0021-9991(05)80016-6

Sahr, K., White, D., & Kimerling, A. J. (2003). *Geodesic Discrete Global Grid Systems*. Cartography and Geographic Information Science, 30(2), 121–134. https://doi.org/10.1559/152304003100011090

Kenner, H. (1976). *Geodesic Math and How to Use It*. University of California Press, Berkeley.

Dutton, G. (1999). *A Hierarchical Coordinate System for Geoprocessing and Cartography*. Springer-Verlag, Berlin.

White, D., Kimerling, A. J., & Song, L. (1998). *Comparing Area and Shape Distortion on Polyhedral-Based Recursive Partitions of the Sphere*. International Journal of Geographical Information Science, 12(8), 805–827.

Fekete, G., & Treinish, L. (1990). *Sphere Quadtrees: A New Data Structure to Support the Visualization of Spherically Distributed Data*. SPIE, Extracting Meaning from Complex Data: Processing, Display, Interaction, 1259, 242–250.

Heikes, R., & Randall, D. A. (1995). *Numerical Integration of the Shallow-Water Equations on a Twisted Icosahedral Grid. Part I: Basic Design and Results of Tests*. Monthly Weather Review, 123(6), 1862–1880.

Heikes, R., & Randall, D. A. (1995). *Numerical Integration of the Shallow-Water Equations on a Twisted Icosahedral Grid. Part II: A Detailed Description of the Grid and an Analysis of Numerical Accuracy*. Monthly Weather Review, 123(6), 1881–1887.

Thuburn, J. (1997). *A PV-Based Shallow-Water Model on a Hexagonal-Icosahedral Grid*. Monthly Weather Review, 125(9), 2328–2347.

Sahr, K., & White, D. (1998). *Discrete Global Grid Systems*. Computing Science and Statistics, 30, 269–278.

Gates, W. L., & Riegel, C. A. (1962). *A Study of Numerical Errors in the Integration of Barotropic Flow on a Spherical Grid*. Journal of Geophysical Research, 67(2), 773–784.

Phillips, N. A. (1959). *Numerical Integration of the Primitive Equations on the Hemisphere*. Monthly Weather Review, 87(9), 333–345.

Kurihara, Y. (1965). *Numerical Integration of the Primitive Equations on a Spherical Grid*. Monthly Weather Review, 93(7), 399–415.

Calhoun, D., Helzel, C., & LeVeque, R. J. (2008). *A Finite Volume Grid for Solving Hyperbolic Problems on the Sphere*. In: Benzoni-Gavage, S., Serre, D. (eds), *Hyperbolic Problems: Theory, Numerics, Applications*, Springer, Berlin.

Steppeler, J., Ripodas, P., Jonkheid, B., & Thomas, S. (2008). *Third-Order Finite-Difference Schemes on Icosahedral-Type Grids on the Sphere*. Monthly Weather Review, 136(7), 2683–2698. https://doi.org/10.1175/2007MWR2224.1

Walko, R. L., & Avissar, R. (2011). *A Direct Method for Constructing Refined Regions in Unstructured Conforming Triangular–Hexagonal Computational Grids: Application to OLAM*. Monthly Weather Review, 139(12), 3923–3937. https://doi.org/10.1175/MWR-D-11-00021.1

Blick, G., & Donnelly, N. (2016). *From static to dynamic datums: 150 years of geodetic datums in New Zealand*. New Zealand Journal of Geology and Geophysics, 59(1), 15–21. https://doi.org/10.1080/00288306.2015.1128451

Osserman, R. (2004). *Mathematical Mapping from Mercator to the Millennium*. In *Mathematical Adventures for Students and Amateurs*, D. F. Hayes & T. Shubin (Eds.), Mathematical Association of America, pp. 233–257. https://www.math.purdue.edu/~eremenko/dvi/osserman-merc.pdf

Pędzich, P. (2005). *Conformal Projection with Minimal Distortions*. Proceedings of the 22nd International Cartographic Conference (ICC2005), A Coruña, Spain. https://icaci.org/files/documents/ICC_proceedings/ICC2005/htm/pdf/oral/TEMA2/Session%202/PAWEL%20PEDZICH2.pdf

Peters, A. (1983). *The New Cartography*. Friendship Press, New York.

Laskowski, P. (1997). *Part 1: Distortion-Spectrum Fundamentals: A New Tool for Analyzing and Visualizing Map Distortions*, and *Part 2: Distortion-Spectrum Applications: Projections Cross-Breeding*. Cartographica, 34(3), Monograph 50, 3–39.

Goldberg, D. M., & Gott, J. R., III (2008). *Flexion and Skewness in Map Projections of the Earth*. Cartographica: The International Journal for Geographic Information and Geovisualization, 42(4), 297–318. https://doi.org/10.3138/carto.42.4.297

Gott, J. R., III, Goldberg, D. M., & Vanderbei, R. J. (2021). *Flat Maps that Improve on the Winkel Tripel*. arXiv:2102.08176 [physics.geo-ph]. https://arxiv.org/abs/2102.08176

PROJ contributors (2026). *Quadrilateralized Spherical Cube* [Software documentation]. PROJ 9.8.1. https://proj.org/en/stable/operations/projections/qsc.html

Chan, F. K., & O'Neill, E. M. (1975). *Feasibility study of a quadrilateralized spherical cube earth data base*. Tech. Rep. EPRF 2-75 (CSC), Computer Sciences Corporation, System Sciences Division, Silver Spring, MD, for the (U.S. Navy) Environmental Prediction Research Facility.

O'Neill, E. M., & Laubscher, R. E. (1976). *Extended studies of a quadrilateralized spherical cube earth data base*. Tech. Rep. EPRF 3-76 (CSC), Computer Sciences Corporation, System Sciences Division, Silver Spring, MD.

Veach, E. (2017). *Announcing the S2 Library: Geometry on the Sphere*. Google Open Source Blog, December 5, 2017. https://opensource.googleblog.com/2017/12/announcing-s2-library-geometry-on-sphere.html

MapTools.com (compiler) (2026). *UTM & MGRS Coordinate System History* [Web archive of scanned 1940s military documents]. https://maptools.com/mgrs_history

Wikipedia contributors (2026). *Military Grid Reference System* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Military_Grid_Reference_System

Vanderhaeghe, F. (2023). *The UTM grid and the MGRS grid: not quite the same* [Tutorial article]. Research Institute for Nature and Forest (INBO) tutorials site. https://tutorials.inbo.be/articles/spatial_mgrs/

Wikipedia contributors (2026). *Web Mercator projection* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Web_Mercator_projection

Battersby, S. E., Finn, M. P., Usery, E. L., & Yamamoto, K. H. (2014). *Implications of Web Mercator and Its Use in Online Mapping*. Cartographica, 49(2), 85–101. https://doi.org/10.3138/carto.49.2.2313

MapAtlas (2026). *What Is Web Mercator? The Map Projection Behind Every Web Map* [Blog post]. MapAtlas Blog. https://mapatlas.eu/blog/what-is-web-mercator

Geography Worlds (2026). *Google Maps Projection Explained: Web Mercator (EPSG:3857)* [Blog post]. https://geographyworlds.com/blog/google-maps-projection-explained/

Cotter, C. H. (1976). *The Early History of Great Circle Sailing*. Journal of Navigation, 29(3), 253–262. https://doi.org/10.1017/S0373463300039114

Library of Congress, Geography and Map Division (2026). *Great Circle Sailing Chart of the North Atlantic Ocean* [Item description, digital collection]. https://www.loc.gov/item/2021668447/

Wikipedia contributors (2026). *Flag of the United Nations* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Flag_of_the_United_Nations

Geography Realm (2023). *The Map Projection of the United Nations' Flag* [Blog post]. https://www.geographyrealm.com/map-projection-united-nations-flag/

International Civil Aviation Organization, Postal History Programme (2026). *The Early Emblems* [Postal-history documentation page]. https://www.icao.int/sites/default/files/postalhistory/the_early_emblems.htm

Bourneuf, P.-E. (2024). *The Fascinating Story of How the UN Emblem Came to Be* [Article citing UN Archives Geneva material]. https://www.linkedin.com/pulse/fascinating-story-how-un-emblem-came-pierre-etienne-bourneuf-idyge

Wikipedia contributors (2026). *Goode homolosine projection* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Goode_homolosine_projection

Wikipedia contributors (2026). *Interruption (map projection)* [Encyclopedia article]. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Interruption_(map_projection)

Esri (2026). *Goode homolosine* [Software documentation]. ArcGIS Pro Documentation. https://doc.esri.com/en/arcgis-pro/latest/help/mapping/properties/goode-homolosine.html

Wikipedia contributors (2026). *Geohash* [Encyclopedia article]. https://en.wikipedia.org/wiki/Geohash

Morton, G. M. (1966). *A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing*. IBM Research Report, IBM Ltd., Ottawa, Canada. March 1, 1966. https://dominoweb.draco.res.ibm.com/0dabf9473b9c86d48525779800566a39.html

Rinckes, D., & Bunge, P. (Google) (2026). *Open Location Code Definition* [Software specification]. https://github.com/google/open-location-code/blob/main/Documentation/Specification/olc_definition.adoc

Rinckes, D. (2015). *Open Location Code: Addresses for everything, everywhere* [Blog post]. Google Open Source Blog. https://opensource.googleblog.com/2015/04/open-location-code-addresses-for.html

Wikipedia contributors (2026). *Open Location Code* [Encyclopedia article]. https://en.wikipedia.org/wiki/Open_Location_Code

Peirce, C. S. (1879). *A Quincuncial Projection of the Sphere*. American Journal of Mathematics, 2(4), 394–396.

Guyou, É. (1887). *Nouveau système de projection de la sphère*. Annales Hydrographiques, 2e série, 9, 16–35.

Wikipedia contributors (2026). *Peirce quincuncial projection* [Encyclopedia article]. https://en.wikipedia.org/wiki/Peirce_quincuncial_projection

Wikipedia contributors (2026). *Guyou hemisphere-in-a-square projection* [Encyclopedia article]. https://en.wikipedia.org/wiki/Guyou_hemisphere-in-a-square_projection

Esri (2026). *Peirce quincuncial* [Software documentation]. ArcGIS Pro Documentation. https://pro.arcgis.com/en/pro-app/latest/help/mapping/properties/peirce-quincuncial.htm

Open Geospatial Consortium (2017). *OGC Abstract Specification Topic 21: Discrete Global Grid Systems Abstract Specification*, v1.0, document 15-104r5. Editor: M. B. J. Purss. https://docs.ogc.org/as/15-104r5/15-104r5.html

Purss, M. B. J., Gibb, R., Samavati, F., & Peterson, P. (2016). *The OGC Discrete Global Grid System Core Standard: A Framework for Rapid Geospatial Integration*. IEEE International Geoscience and Remote Sensing Symposium (IGARSS) proceedings.

Federal Office of Topography swisstopo (2026). *Swiss reference systems: CHTRS95 and CH1903+* [Technical documentation page]. https://www.swisstopo.admin.ch/en/swiss-reference-systems

Federal Office of Topography swisstopo (2026). *Swiss Grid and UTM – Map projection* [Technical documentation page]. https://www.swisstopo.admin.ch/en/swiss-map-projections

International Association of Oil & Gas Producers, EPSG Geodetic Parameter Dataset (2026). *CH1903+ / LV95 — EPSG:2056*. https://epsg.io/2056

National Geospatial-Intelligence Agency, Office of Geomatics (2026). *NGA Geomatics – Coordinate Systems* [Official government documentation page]. https://earth-info.nga.mil/index.php?dir=coordsys&action=coordsys

*Life Presents R. Buckminster Fuller's Dymaxion World* [Photographic essay]. Life magazine, March 1, 1943.

Boston Rare Maps (2026). *The Dymaxion Airocean World Map, by R. Buckminster Fuller* [Antiquarian map dealer listing, provenance essay]. https://bostonraremaps.com/inventory/buckminster-fuller-dymaxion-airocean-world-1954/

National Geographic Society, Education division (2026). *Selecting a Map Projection* [Educational resource page]. https://education.nationalgeographic.org/resource/selecting-map-projection/

Lapaine, M. (2024). *Standard Points and Lines in Map Projections*. KN – Journal of Cartography and Geographic Information. https://link.springer.com/article/10.1007/s42489-024-00168-8

van Wijk, J. J. (2008). *Unfolding the Earth: Myriahedral Projections*. The Cartographic Journal, 45(1), 32–42. https://vanwijk.win.tue.nl/myriahedral/CAJ103.pdf

Spilhaus, A. (1983). *World Ocean Maps: The Proper Places to Interrupt*. Proceedings of the American Philosophical Society, 127(1), 50–60.

Gauss, C. F. (1825). *Allgemeine Auflösung der Aufgabe: die Theile einer gegebnen Fläche auf einer andern gegebnen Fläche so abzubilden, daß die Abbildung dem Abgebildeten in den kleinsten Theilen ähnlich wird*. Astronomische Abhandlungen, Altona.

Voss, K. (2016). *Euler and Chebyshev: From the Sphere to the Plane and Backwards*. arXiv:1608.02724. https://arxiv.org/pdf/1608.02724

Krüger, J. H. L. (1912). *Konforme Abbildung des Erdellipsoids in der Ebene*. Veröffentlichung des Königlich Preußischen Geodätischen Institutes, Neue Folge, Nr. 52. B. G. Teubner, Leipzig. DOI: 10.2312/GFZ.b103-krueger28.

Hunger, Fritz (1985). *Geschichte des Deutschen Vereins für Vermessungswesen (DVW), Teil II: 1918/19–1945/50. Zeitschrift für Vermessungswesen* (ZfV), Sonderheft 23/1985, 110. Jg., Wittwer-Verlag, Stuttgart, pp. 75–136.

Vermessungsgeschichte.de, redaktion (2003). *Gauß-Krüger-Koordinaten: Einführung vor 80 Jahren empfohlen*, citing *Zeitschrift für Vermessungswesen* (ZfV) Sonderheft 23 (August 1985), p. 87. https://www.vermessungsgeschichte.de/detail/gauss-krueger-koordinaten-einfuehrung-vor-80-jahren-empfohlen

Uber Technologies, Inc. (2026). *H3: Hexagonal hierarchical geospatial indexing system* [Source code]. https://github.com/uber/h3